APMA 2580

Multiscale Computational Fluid Dynamics

Instructor: **Professor George Karniadakis**
Location: **170 Hope Street, Room 108**
Time: **Tuesday 10:30 - 1:00 pm**

Today's topic:

Bottom-Up Models for CFD

Part I: Molecular Dynamics (MD)

Part II: Dissipative Particle Dynamics (DPD)

By **Zhen Li**
Email: zhen_li@brown.edu

May 3, 2016

Part I: Molecular Dynamics

- **Basics of MD simulation**
  - motivation, history, typical length and time scales
- **Potentials**
  - Non-bonded interactions
  - Bonded interactions
- **Algorithms for time integration**
  - Verlet, velocity-Verlet, Beeman
- **How do you practically run a MD simulation**
  - Scaling, Periodic BC, Potential cut-offs, cell-list and Verlet list, thermostats
- **Analysis of MD**
  - Configurations, time correlations, transport properties

Continuum mechanics vs. atomistic viewpoint

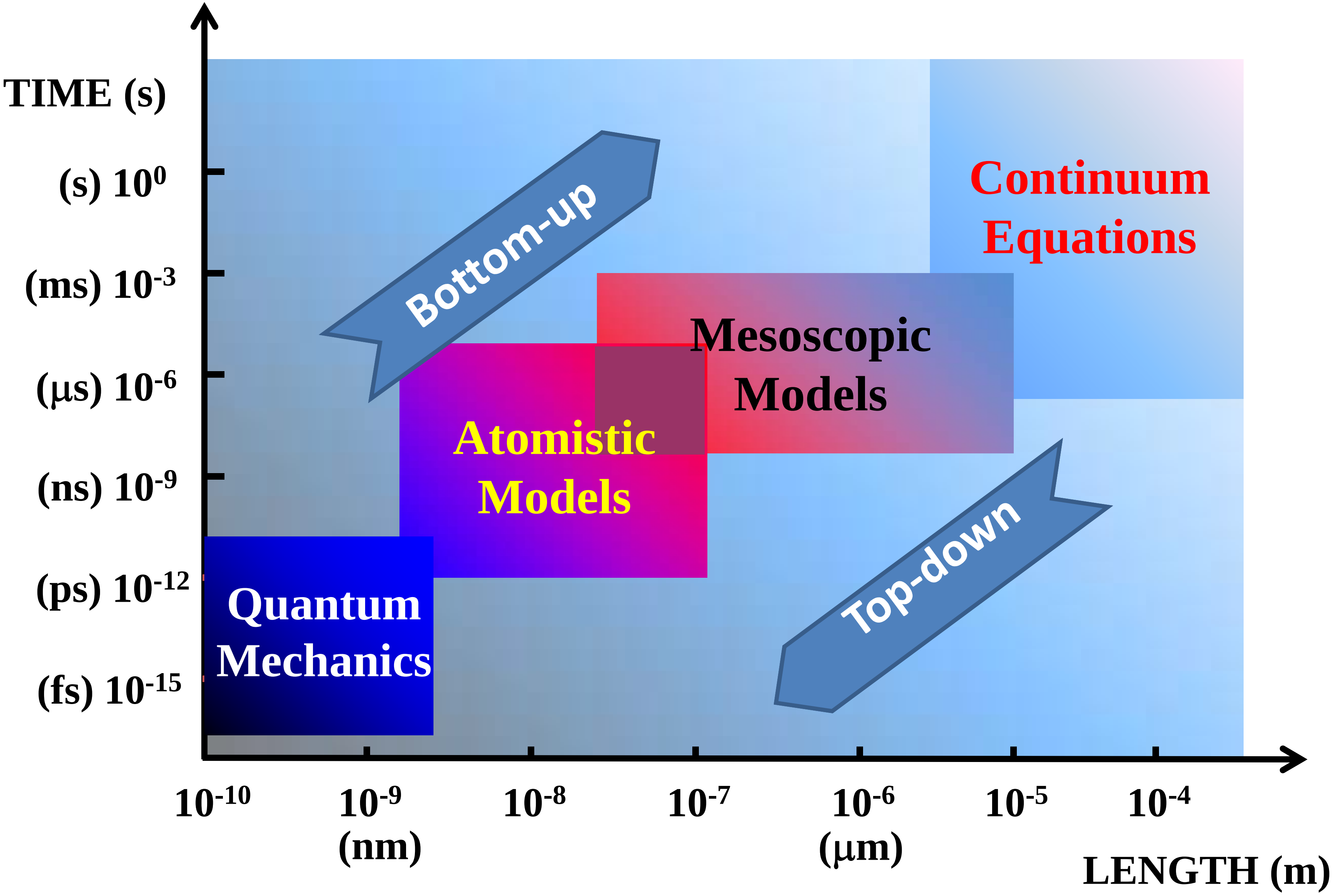
Continuum assumption: (PDEs)

- Material can be modeled as a continuous mass that fills the entire region of space it occupies.
- No underlying inhomogeneous microstructure, that is, matter can be divided infinitely without change of material properties.
- It ignores the fact that matter is made of atoms.
- Only valid on length scales much greater than that of inter-atomic distances.

Atomistic viewpoint: (Newton's second law F=ma)

- Material is made of discrete atoms.
- No spatial discretization necessary – given by atomic distances
- More fundamental description of the world, does not distinguish different subjects (physics, chemistry, biology, material science, et. al.)

Length and time scales for different models



D. G. Vlachos, Adv. Chem. Eng., 2005, **30**: 1-61.

Basics of MD simulation

If all of scientific knowledge were to be destroyed, and only one sentence passed on to the next generations of creatures, what statement would contain the most information in the fewest words?

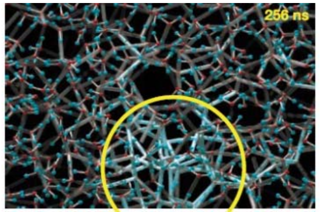
Richard Feynman: "I believe it is the *atomic hypothesis* (or the atomic *fact*, or whatever you wish to call it) that *all things are made of atoms*."

---- *The Feynman Lectures on Physics*

Motivation of MD simulation

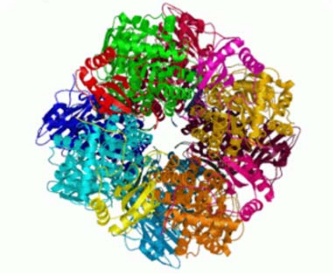
The computer experiments (*in-silico* experiments):

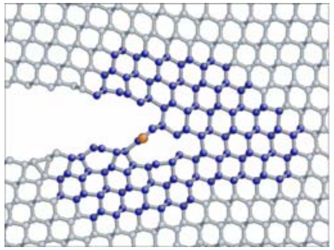
- For a better understanding of underlying mechanisms of real experiments



2002, Molecular dynamics simulation of the ice nucleation and growth process leading to water freezing,
M Matsumoto, S Saito and I Ohmine
Nature **416**, 409-413

- Allow to study the dynamic processes at atomistic level
  - Dynamical events control processes which affect functional properties of the biomolecules.
  - Fast dynamic processes of crack development

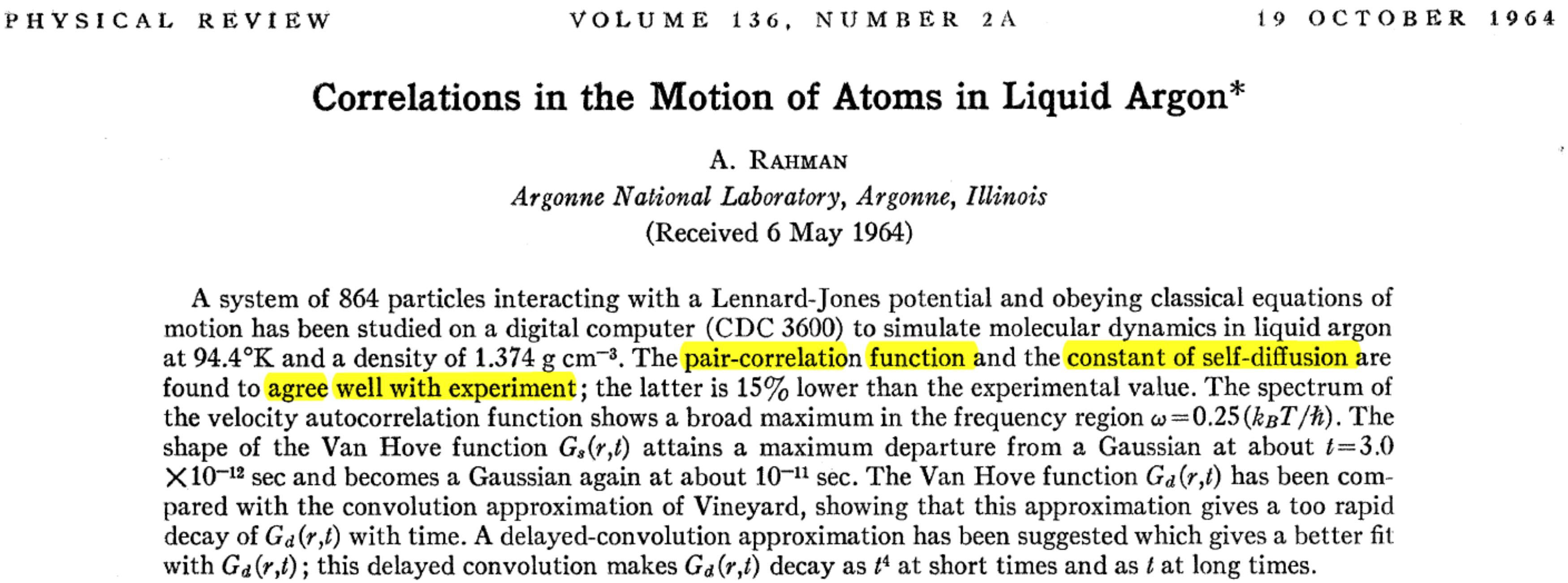




Basics of MD simulation

History

Rahman published a landmark simulation, establishing the field.



PHYSICAL REVIEW VOLUME 136, NUMBER 2A 19 OCTOBER 1964

Correlations in the Motion of Atoms in Liquid Argon*

A. RAHMAN

Argonne National Laboratory, Argonne, Illinois

(Received 6 May 1964)

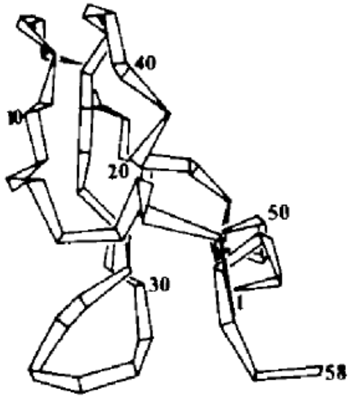
A system of 864 particles interacting with a Lennard-Jones potential and obeying classical equations of motion has been studied on a digital computer (CDC 3600) to simulate molecular dynamics in liquid argon at 94.4°K and a density of 1.374 g cm^{-3}. The pair-correlation function and the constant of self-diffusion are found to agree well with experiment; the latter is 15% lower than the experimental value. The spectrum of the velocity autocorrelation function shows a broad maximum in the frequency region $\omega = 0.25(k_BT/\hbar)$. The shape of the Van Hove function $G_s(r,t)$ attains a maximum departure from a Gaussian at about $t=3.0\times10^{-12}$ sec and becomes a Gaussian again at about 10^{-11} sec. The Van Hove function $G_d(r,t)$ has been compared with the convolution approximation of Vineyard, showing that this approximation gives a too rapid decay of $G_d(r,t)$ with time. A delayed-convolution approximation has been suggested which gives a better fit with $G_d(r,t)$; this delayed convolution makes $G_d(r,t)$ decay as t^4 at short times and as t at long times.

Only 864 atoms,

System properties, structure of atoms and coefficient of self-diffusion, compared well with experimental data

History

1975, Computer simulation of protein folding
M Levitt, A Warshel
Nature 253:94.
(~750 atoms)
1979, Dynamics of ligand binding to heme protein
DA Case, M Karplus
J Mol Biol 132:343



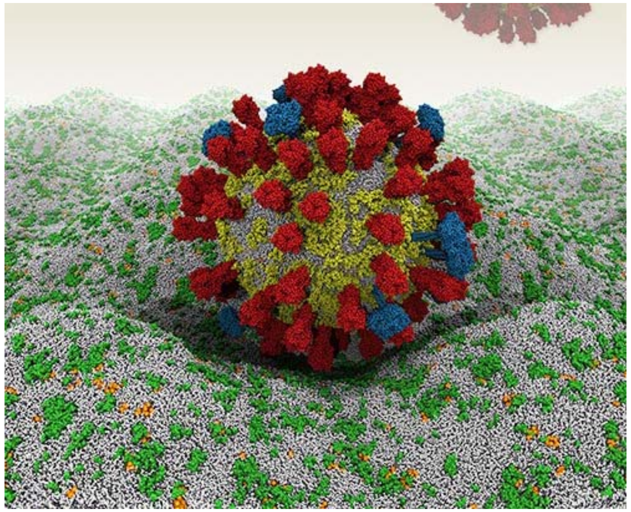
M. Levitt, A. Warshel together with M. Karplus received the 2013 Nobel Prize in Chemistry awarded in part for the application of MD to proteins.

.

.

.

.



2015, Nothing to Sneeze At: A Dynamic and Integrative Computational Model of an Influenza A Virion
T Reddy, D Shorthouse, DL Parton, E Jefferys, et al.
Structure 23, 584–597
(> 10 M particles**)**

Typical length and time scales of MD systems

Length scale:

Limited by the number of atoms that can be included in the simulation.

Typical MD systems contain thousands to millions of atoms.

System size: several nanometres to hundreds of nanometres.

$$(1\,nm\ = 10^{-9}m\,)$$

Time scale:

Time step is limited by the smallest oscillation period of fastest atomic motions.

Simulation times: picoseconds to nanoseconds.

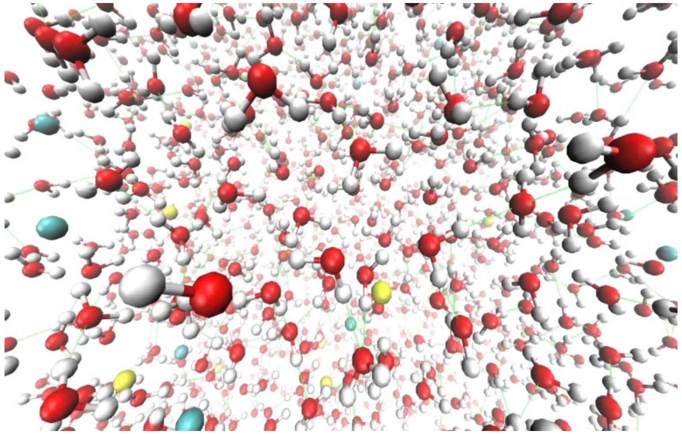
$$(1\,ps\ = 10^{-12}s, 1\,ns\ = 10^{-9}s)$$

$$f \sim 100THz = \ 10^{14}s^{-1}$$

Basics of MD simulation

A MD system is made of many atoms

- **N** particles
- Mass m_i
- Position $\boldsymbol{r}_i$
- Velocity $\boldsymbol{v}_i$



Total energy of system

$$\mathrm{E} = \mathrm{K} + \mathrm{V}$$

$$= \frac{1}{2}\sum_{i=1}^{N} m_i {\boldsymbol{v}_i}^2 + \sum_{i=1}^{N} V(\boldsymbol{r}_i)$$

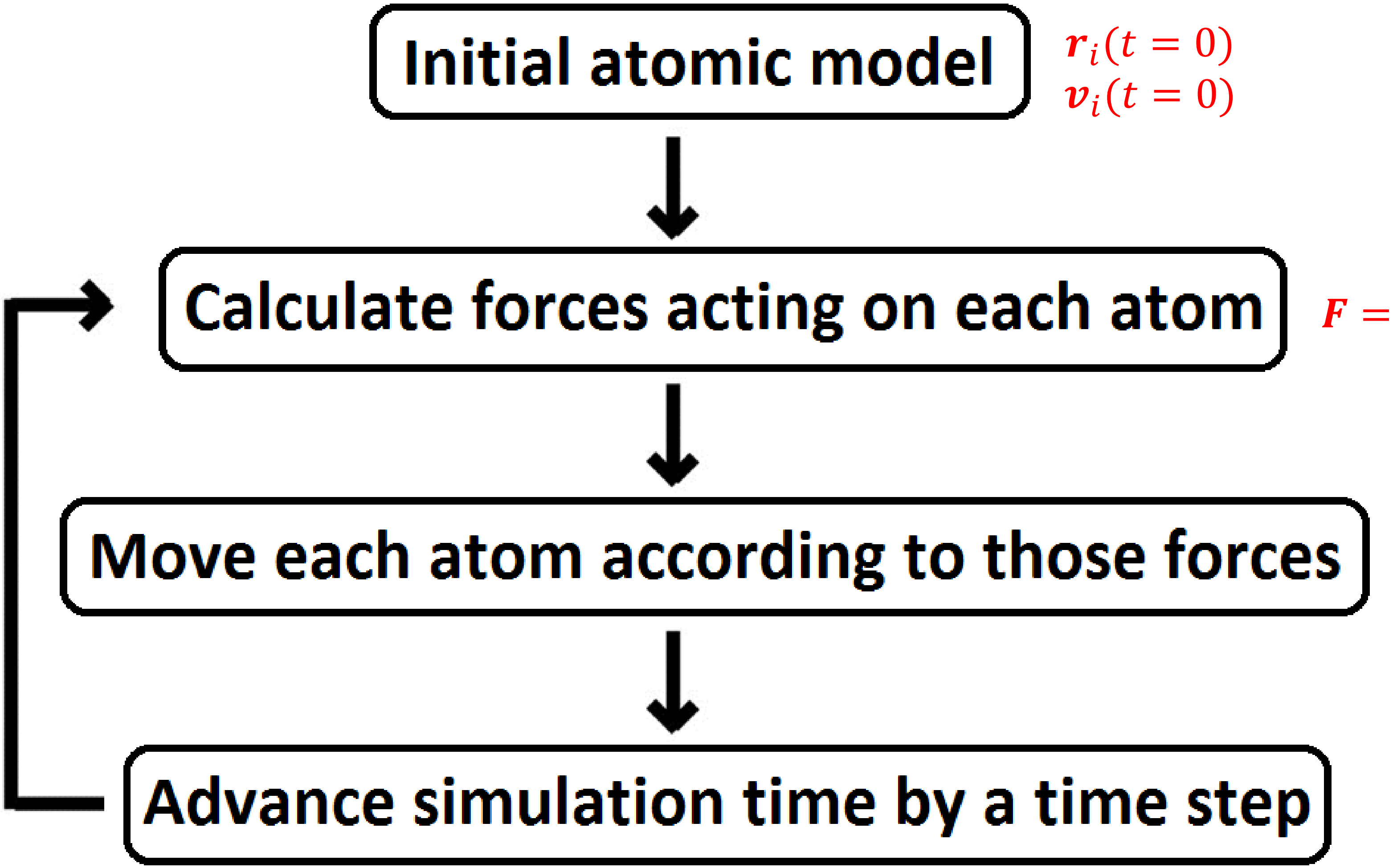
Coupled system of N-body problem, no exact solution for N > 2

Equation of motion for atoms

$$m_i \frac{\mathrm{d}^2\mathbf{r}_\mathrm{i}}{\mathrm{dt}^2} = -\nabla_{r_i}\mathrm{V}(\boldsymbol{r}_i)$$

- System of coupled 2nd order nonlinear differential equations
- Solved by discretizing in time (spatial discretization given by individual atoms)

Procedure of MD simulation



Initial atomic model

$$\boldsymbol{r}_i(t=0)$$
$$\boldsymbol{v}_i(t=0)$$

Calculate forces acting on each atom

$$\boldsymbol{F} = -\nabla V(\boldsymbol{r}_i)$$

Move each atom according to those forces

$$\frac{d\boldsymbol{r}_i}{dt} = \boldsymbol{v}_i$$
$$m\frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{F}_i$$

Advance simulation time by a time step

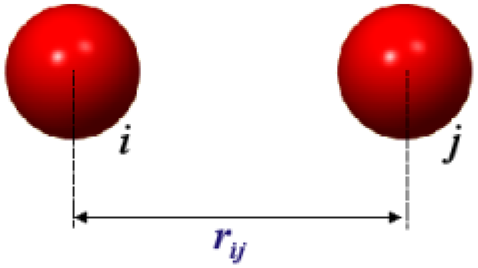
Part I: Molecular Dynamics

- Basics of MD simulation
  - motivation, history, typical length and time scales
- **Potentials**
  - Non-bonded interactions
  - Bonded interactions
- Algorithms for time integration
  - Verlet, velocity-Verlet, Beeman
- How do you practically run a MD simulation
  - Scaling, Periodic BC, Potential cut-offs, cell-list and Verlet list, thermostats
- Analysis of MD
  - Configurations, time correlations, transport properties

Potentials

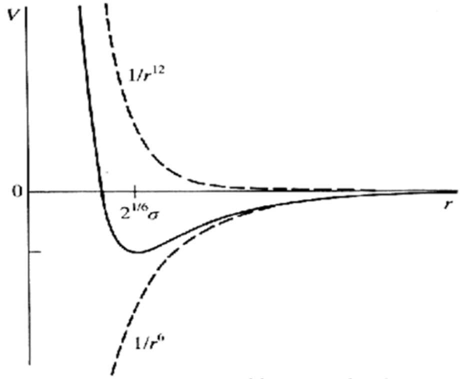
Non-Bonded interactions

Van der Waals interaction is referred to as the combination of attractive and repulsive forces between two atoms, which are not bonded to each other.



Lennard-Jones potential:

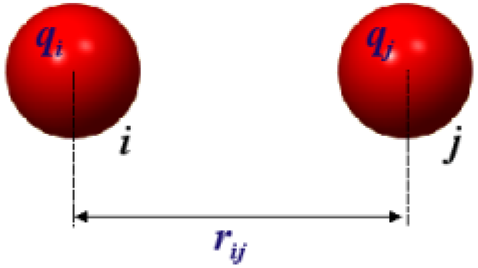
$$V_{LJ} = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right]$$



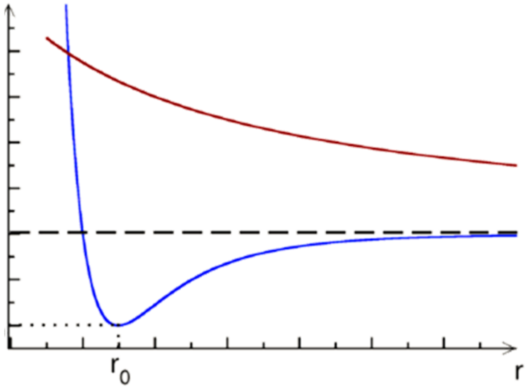
The attractive part (power 6) has been experimentally validated.
The repulsive part (power 12) is empirical.

Lorentz-Berthelot mixing rules: $\sigma_{ij} = \frac{1}{2}(\sigma_{ii} + \sigma_{jj}), \epsilon_{ij} = \sqrt{\epsilon_{ii}\epsilon_{jj}}$

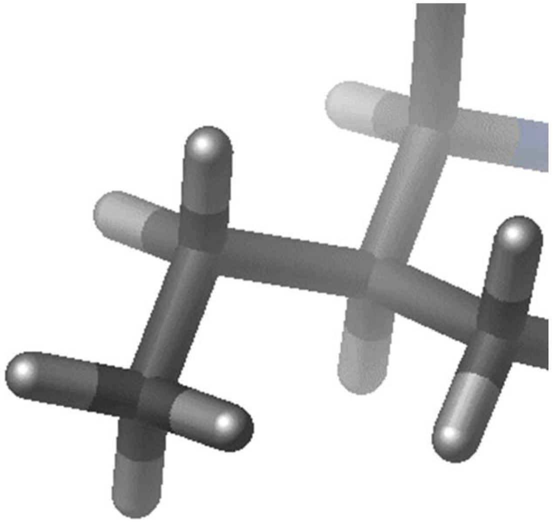
Coulomb interaction is the electrostatic forces between two atoms.

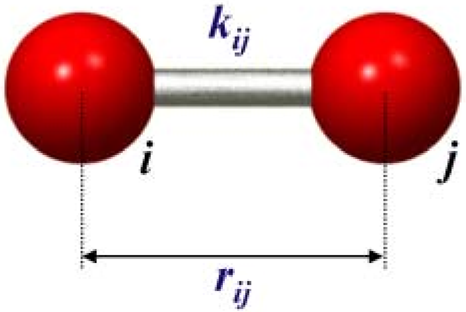


$$V_E = \frac{q_i q_j}{4\pi\varepsilon_{ij} r_{ij}}$$

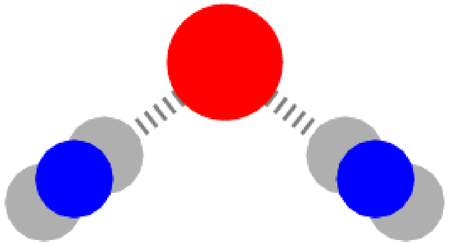


Bonded interactions 1: Bond stretching





$$V_{bond} = K_b(r_{ij} - r_0)^2$$



Vibration of O-H bond

Bond length: $r_0 = 0.958\text{Å}$

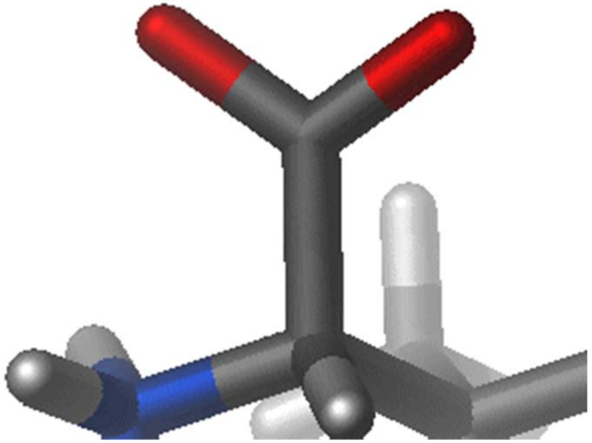
Frequency: $f = 101.9\ \text{THz} \approx 10^{14}s^{-1}$

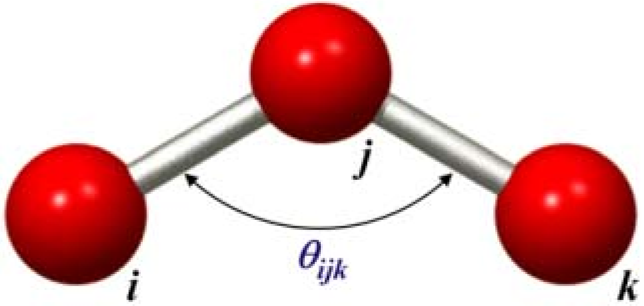
Period: $T \approx 10^{-14}s$

$(\text{Å} = 10^{-10}m, nm = 10^{-9}m)$

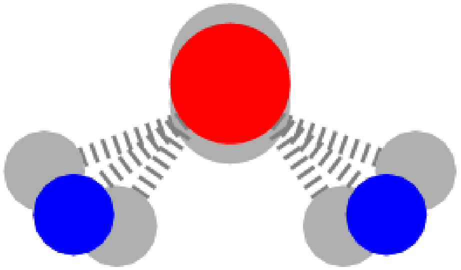
$(fs = 10^{-15}s, ps = 10^{-12}s, ns = 10^{-9}s)$

Bonded interactions 2: Bond angle bending





$$V_{angle} = K_\theta(\theta_{ijk} - \theta_0)^2$$



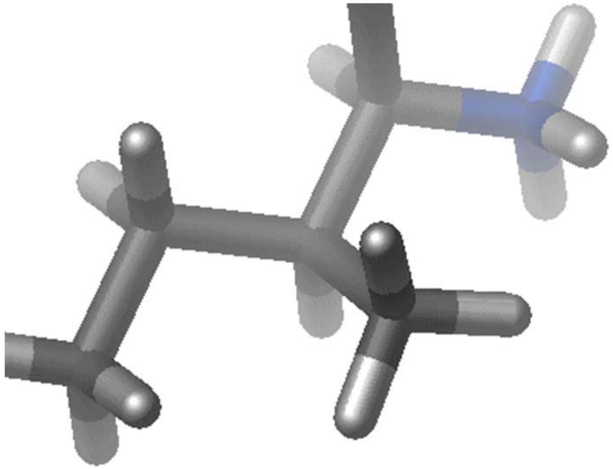
Vibration of O-H angle bending

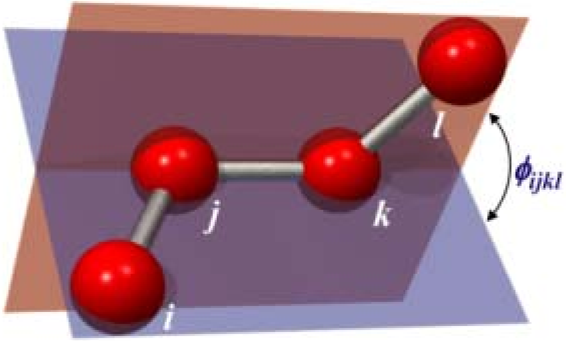
Bond length: $\theta_0 = 104.45°$

Frequency: $f = 47.8 \text{ THz} \approx 5 \times 10^{13} s^{-1}$

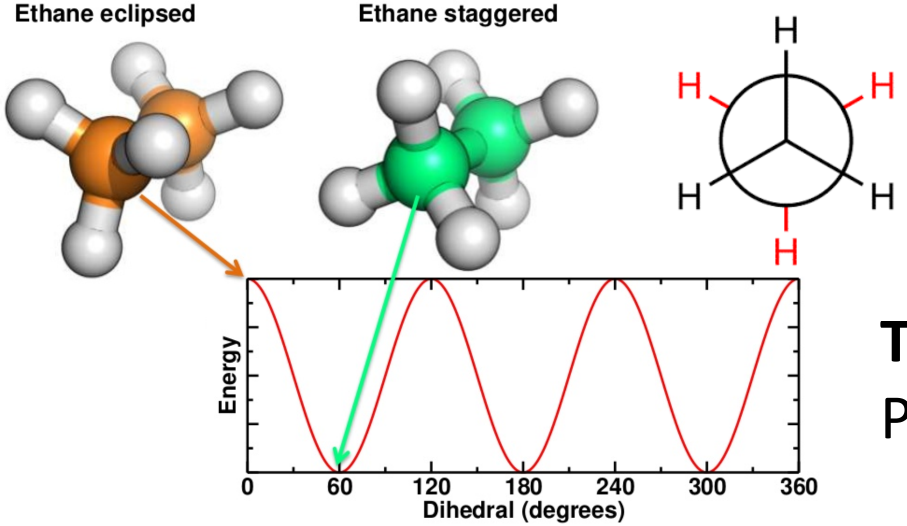
Period: $T \approx 2 \times 10^{-14} s$

Bonded interactions 3 : Proper dihedral angle bending (Torsion)





$$V_{torsion} = K_{\phi}[1+\cos(n\phi_{ijkl})]$$

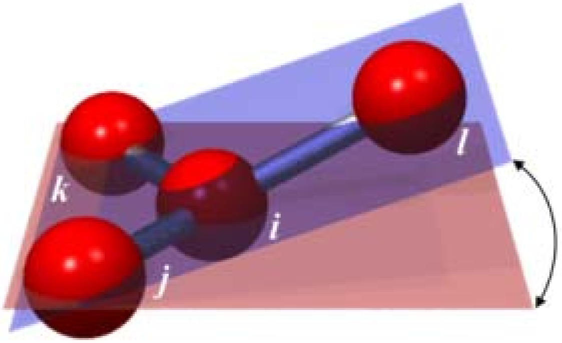


Torsion:

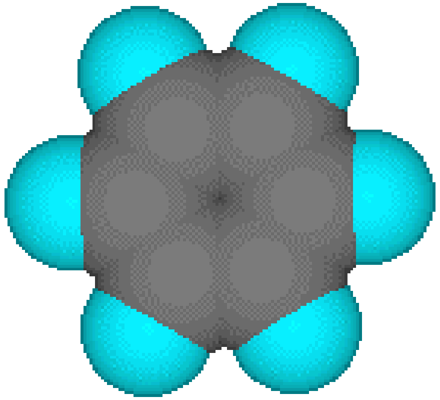
Periodicity needs to be enforced.

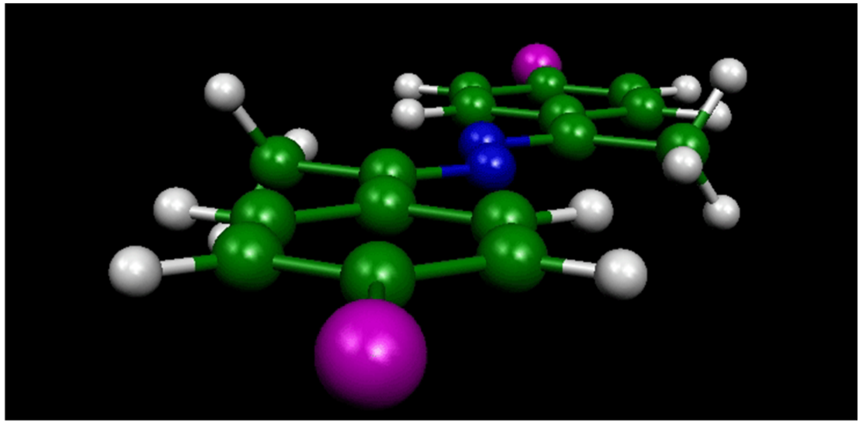
Potentials

Bonded interactions 4 : Improper dihedral angle bending



$$V_{dihedral} = K_{\omega}(\omega - \omega_0)^2$$





Potentials

$$
\begin{aligned}
\mathrm{V}(\mathrm{r_i} \ldots \mathrm{r_N}) = & \sum_{ij} 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] \\
& + \sum_{ij} \frac{q_i q_j}{4\pi\varepsilon_{ij} r_{ij}} \\
& + \sum_{bond} K_b\left(r_{ij} - r_0\right)^2 \\
& + \sum_{angle} K_\theta\left(\theta_{ij} - \theta_0\right)^2 \\
& + \sum_{torsion} K_\phi[1 + \cos(n\phi_{ijkl})] \\
& + \sum_{diheral} K_\omega\left(\omega_{ij} - \omega_0\right)^2
\end{aligned}
$$

Part I: Molecular Dynamics

- Basics of MD simulation
  - motivation, history, typical length and time scales
- Potentials
  - Non-bonded interactions
  - Bonded interactions
- **Algorithms for time integration**
  - Verlet, velocity-Verlet, Beeman
- How do you practically run a MD simulation
  - Scaling, Periodic BC, Potential cut-offs, cell-list and Verlet list, thermostats
- Analysis of MD
  - Configurations, time correlations, transport properties

Algorithms for time integration

Integrator: Verlet Algorithm

Taylor expansion for particle position at $t + \Delta t$:

$$r(t+\Delta t) = r(t) + v(t)\Delta t + \frac{1}{2}\Delta t^2 a(t) + O(\Delta t^3)$$

Similarly, the old position at $t - \Delta t$:

$$r(t-\Delta t) = r(t) - v(t)\Delta t + \frac{1}{2}\Delta t^2 a(t) - O(\Delta t^3)$$

Combine the above two equation, we have

$$r(t+\Delta t) = 2r(t) - r(t-\Delta t) + \Delta t^2 a(t) + O(\Delta t^3)$$

Thus the velocity at t is:

$$v(t) = \dot{r}(t) = \frac{1}{2\Delta t}(r(t+\Delta t) - r(t-\Delta t)) + O(\Delta t^2)$$

- Velocities not explicitly solved.
- A two-step method
- Advantages: simplicity and good stability
- Global error $O(\Delta t^2)$

Algorithms for time integration

Integrator: velocity-Verlet Algorithm

Taylor expansion for particle position and velocity at $t + \Delta t$:

$$r(t+\Delta t) = r(t) + v(t)\Delta t + \frac{1}{2}\Delta t^2 a(t) + O(\Delta t^3)$$

$$v(t+\Delta t) = v(t) + a(t)\Delta t + \frac{1}{2}\dot{a}(t)\Delta t^2 + O(\Delta t^3)$$

Taylor expand acceleration

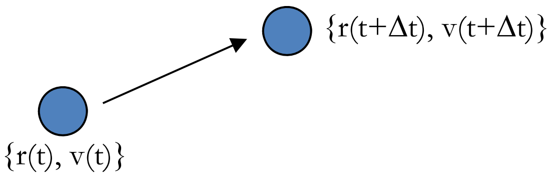
$$\dot{a}(t)\Delta t^2 = \left(\frac{a(t+\Delta t)-a(t)}{\Delta t}\right)\Delta t^2 + O(\Delta t^3)$$

Thus the position and velocity are updated by:

$$r(t+\Delta t) = r(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 + O(\Delta t^3)$$

$$v(t+\Delta t) = v(t) + \frac{1}{2}\Delta t(a(t+\Delta t) + a(t) + O(\Delta t^3)$$

- Velocity calculated explicitly
- Possible to control the temperature
- Stable in long time simulation
- Most commonly used algorithm



Algorithms for time integration

Each integration cycle using velocity-Verlet Algorithm

$$\mathbf{r}_i(t+\Delta t)=\mathbf{r}_i(t)+\Delta t\mathbf{v}_i(t)+\tfrac{1}{2}(\Delta t)^2\mathbf{f}_i(t),$$

$$\tilde{\mathbf{v}}_i(t+\Delta t)=\mathbf{v}_i(t)+\tfrac{1}{2}\Delta t\mathbf{f}_i(t),$$

$$\mathbf{f}_i(t+\Delta t)=\mathbf{f}_i(\mathbf{r}(t+\Delta t),\tilde{\mathbf{v}}(t+\Delta t)),$$

$$\mathbf{v}_i(t+\Delta t)=\mathbf{v}_i(t)+\tfrac{1}{2}\Delta t(\mathbf{f}_i(t)+\mathbf{f}_i(t+\Delta t)).$$

Beeman Algorithm

$$r(t+\Delta t)=r(t)+v(t)\Delta t+\frac{1}{6}\big(4a(t)-a(t-\Delta t)\big)\Delta t^2+O(\Delta t^3)$$

$$v(t+\Delta t)=v(t)+\frac{1}{6}\big(2a(t+\Delta t)+5a(t)-a(t-\Delta t)\big)\Delta t+O(\Delta t^3)$$

Part I: Molecular Dynamics

- Basics of MD simulation
  - motivation, history, typical length and time scales
- Potentials
  - Non-bonded interactions
  - Bonded interactions
- Algorithms for time integration
  - Verlet, velocity-Verlet, Beeman
- **How do you practically run a MD simulation**
  - Scaling, Periodic BC, Potential cut-offs, cell-list and Verlet list, thermostats
- Analysis of MD
  - Configurations, time correlations, transport properties

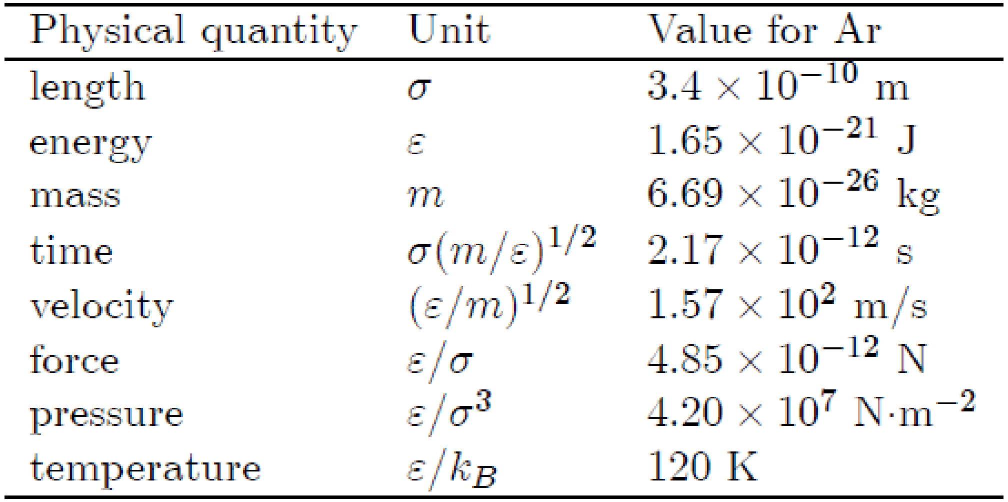
How do you practically run a MD simulation?

1. Parameterization of MD system

- Scaling by model parameters
  - Length scale: size of atom σ
  - Energy scale: depth of LJ potential well ε
  - Mass scale : mass of the atom m
- Other units can be determined by σ, ε and m
  - Time unit: $\tau = \sigma(m/\varepsilon)^{1/2}$
  - Velocity unit: $v = (m/\varepsilon)^{1/2}$
  - Force unit: $F = \varepsilon/\sigma$
  - Pressure unit: $P = \varepsilon/\sigma^3$
  - Temperature unit: $T = \varepsilon/k_B$

An example: liquid argon at 120K:

We choose units of mass, length and energy, then the reduced LJ units become $\sigma = 1, m = 1, \varepsilon = 1.$



Physical quantity	Unit	Value for Ar
length	σ	3.4×10^{-10} m
energy	ε	1.65×10^{-21} J
mass	m	6.69×10^{-26} kg
time	$\sigma(m/\varepsilon)^{1/2}$	2.17×10^{-12} s
velocity	$(\varepsilon/m)^{1/2}$	1.57×10^{2} m/s
force	ε/σ	4.85×10^{-12} N
pressure	ε/σ^3	4.20×10^{7} N$\cdot$m^{-2}
temperature	ε/k_B	120 K

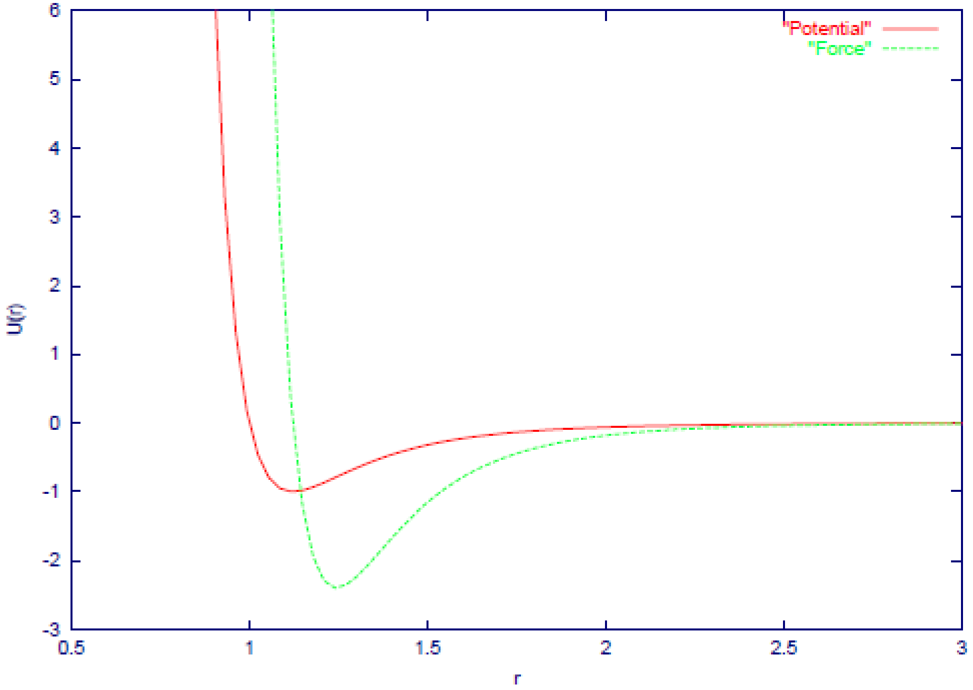
How do you practically run a MD simulation?

1. Parameterization of MD system

Lennard-Jones potential energy function

$$U(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

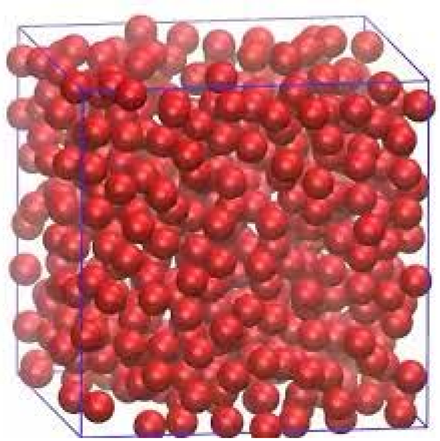
$$F(r) = -\frac{dU(r)}{dr} = \frac{24\varepsilon}{\sigma}\left[2\left(\frac{\sigma}{r}\right)^{13} - \left(\frac{\sigma}{r}\right)^{7}\right]$$

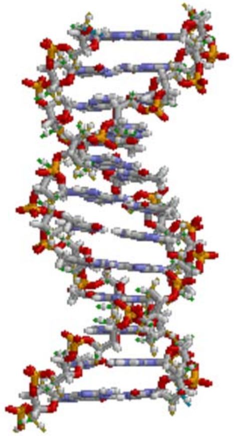


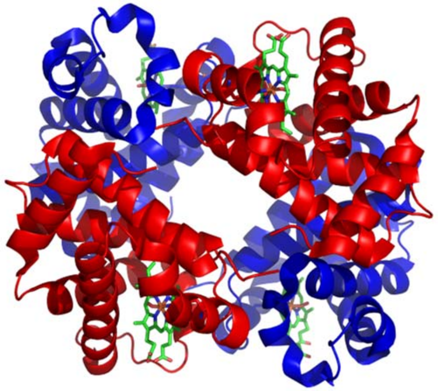
How do you practically run a MD simulation?

2. Get the initial configuration

Simple systems: initial position of particles can be randomly initialized.
Complex systems: taken from data banks, for example **Protein Data Bank** (www.rcsb.org)







3. Assign initial velocities

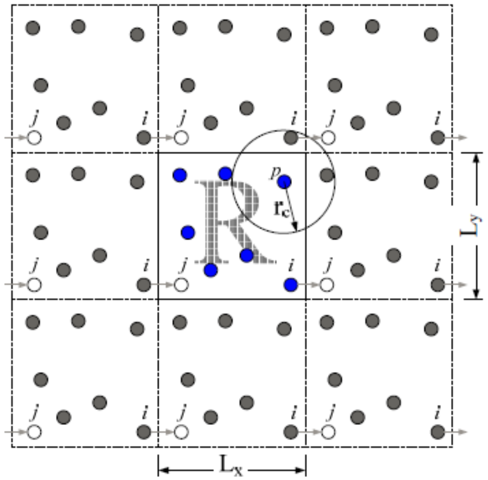
At thermal equilibrium, the mean kinetic energy of the system at temperature T is:

$$\langle K \rangle = \frac{1}{2}\sum_{i=1}^{3N} m_i v_i^2 = \frac{1}{2}(3N)k_B T$$

This can be obtained by assigning the velocity components v_i from a random Gaussian distribution with zero mean and standard deviation $(k_B T/m_i)$.

How do you practically run a MD simulation?

4. Apply periodic boundary conditions



For computational box $0 < x < L$

If $x < 0$ then $x = x + L$
If $x > L$ then $x = x - L$

Minimum image criterion: among all images of a particle, consider only the closest and neglect the rest.

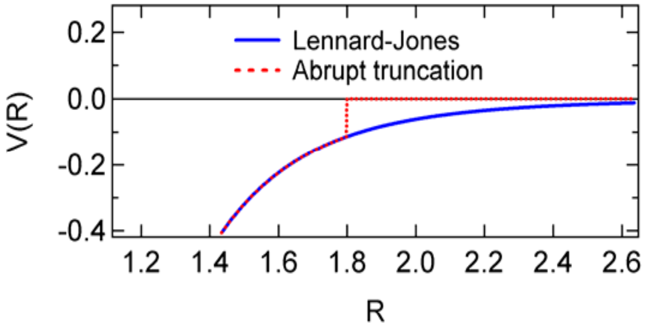
5. Truncate non-bonded interactions

Bonded interactions: local, therefore the computational cost is O(N).

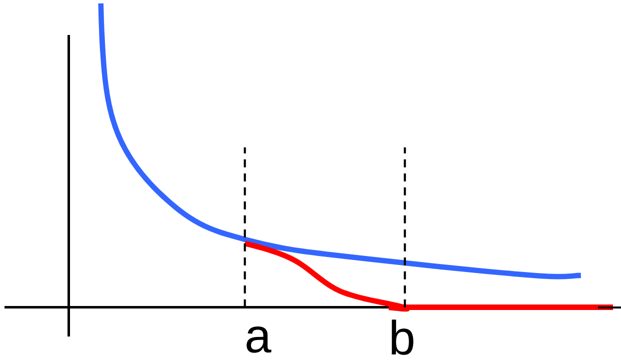
Non-bonded interactions: involve all pairs of atoms, therefore the cost is O(N^2).

Reducing the computing cost: use of cut-off for non-bonded interactions.
The cutoff distance may be no greater than ½ L (L= box length)

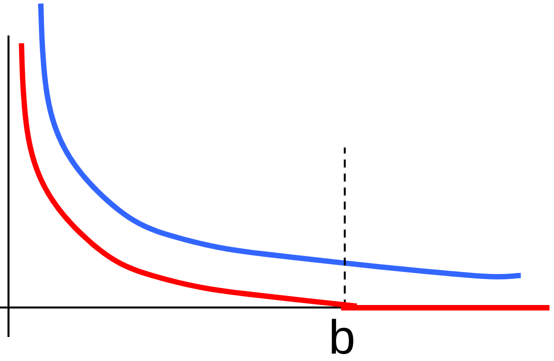
1. Abrupt truncation



2. Switching



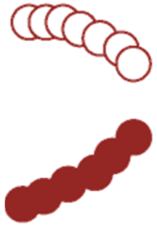
3. Shifting



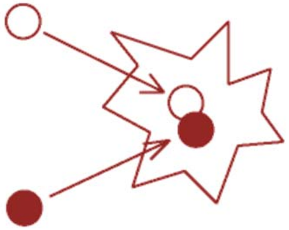
How do you practically run a MD simulation?

6. Choosing the time step

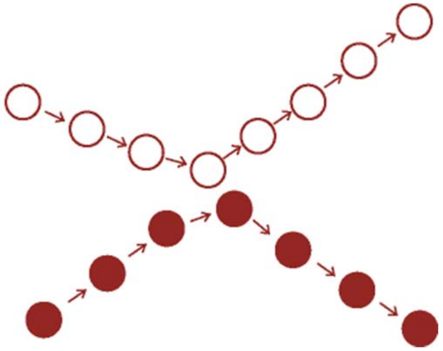
- Too small: covering small conformation space



- Too large: numerical instability



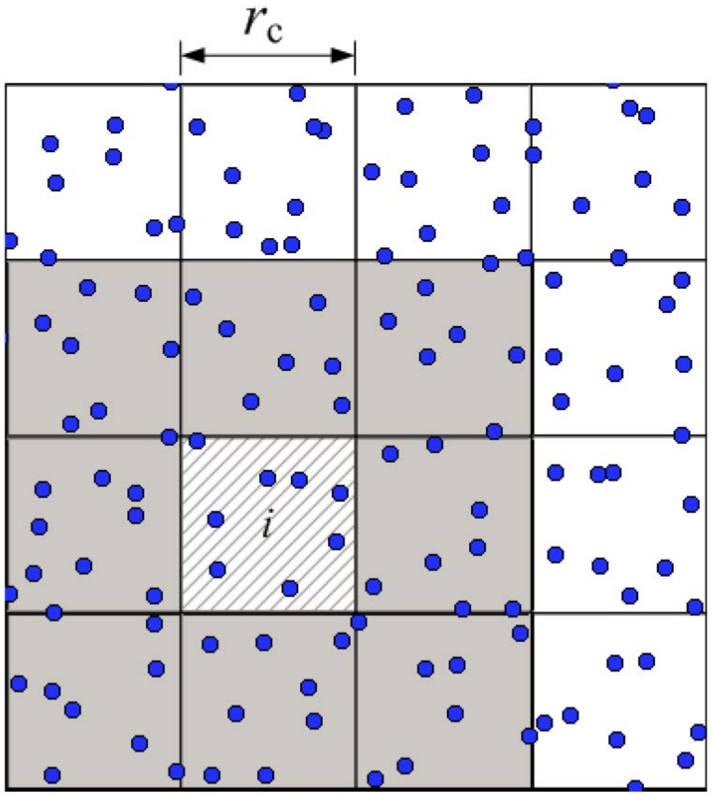
- Suggested time steps
  - Translation, 10 fs
  - Flexible molecules and rigid bonds, 2fs
  - Flexible molecules and bonds, 1fs



How do you practically run a MD simulation?

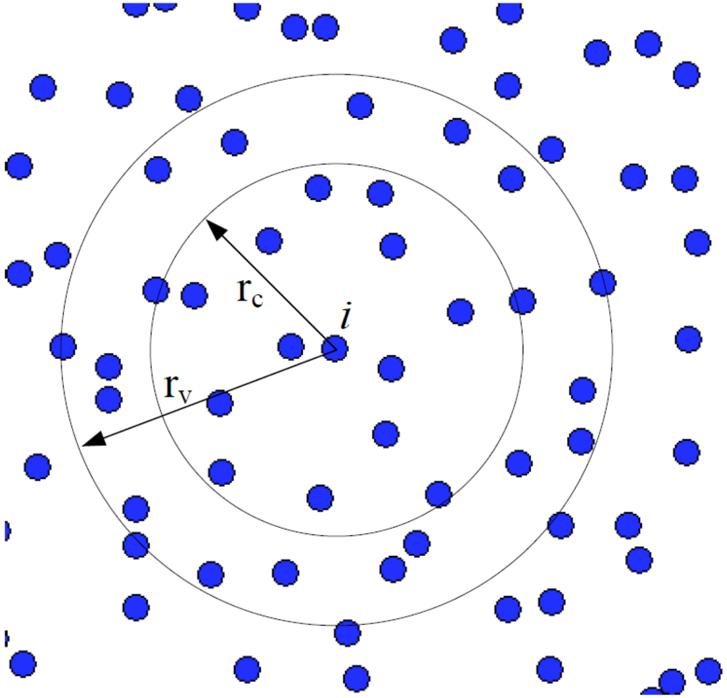
7. Saving CPU time

Cell list



Update cell list at every time steps
$Cell_i = [r_i/r_c]$

Verlet list



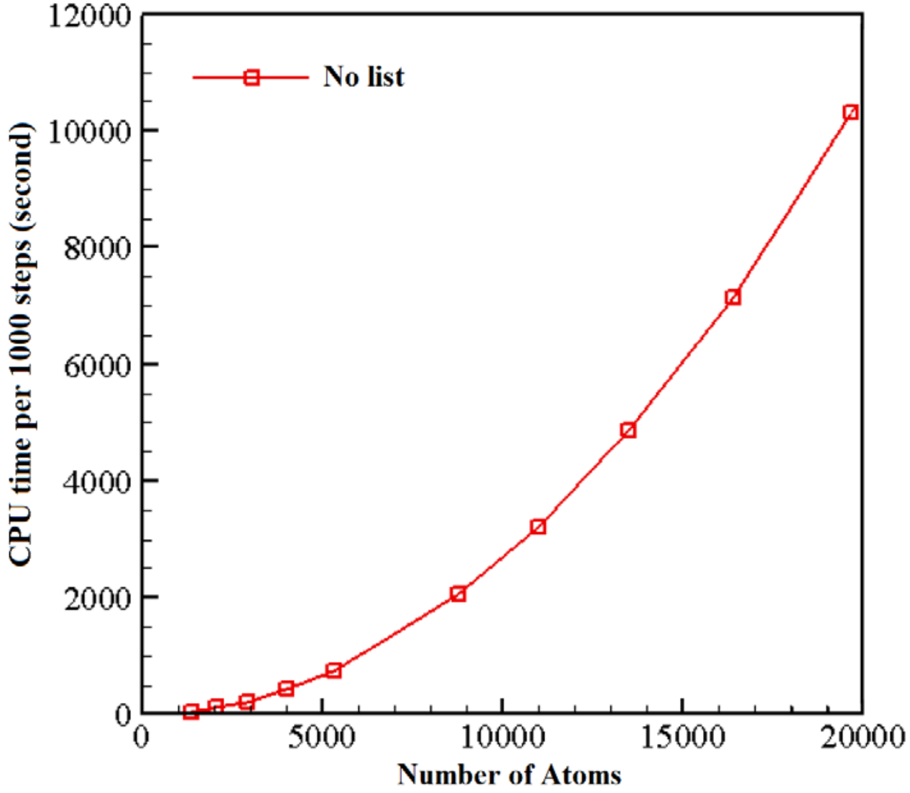
Update verlet list only when
$|\Delta r|_{max} > r_v - r_c$

How do you practically run a MD simulation?

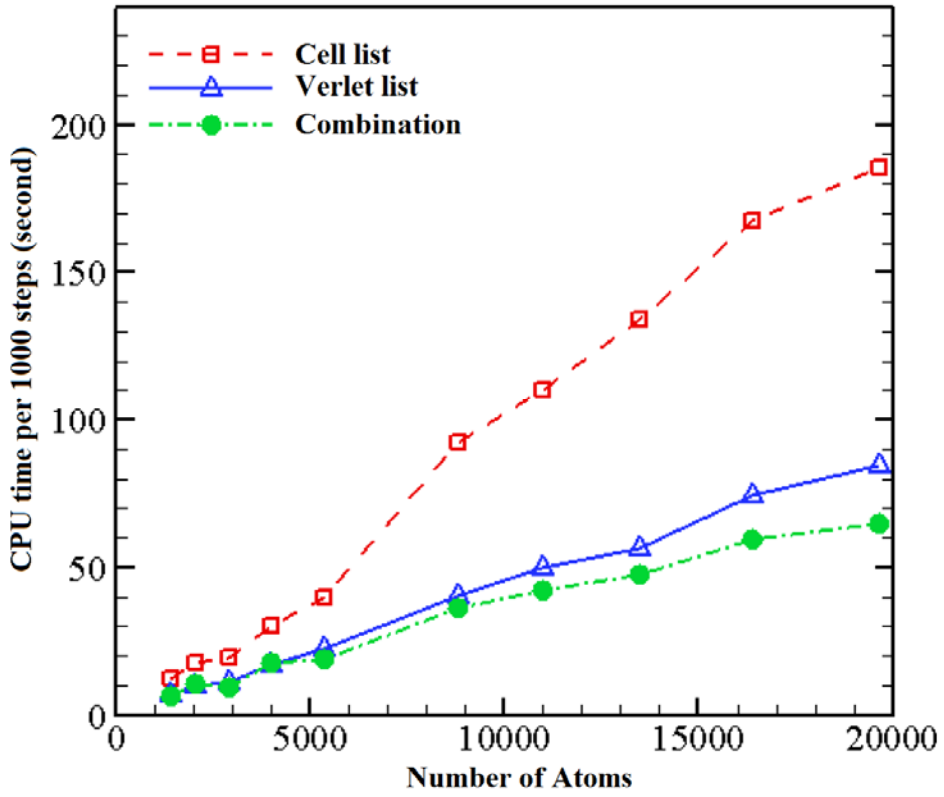
7. Saving CPU time

An example:

Computational Cost : *O(N^2)*



With Lists, computational cost: *O(N)*



Long-range electrostatic interactions O(N^2):

- Ewald summation (Ewald, 1921): ~O($N^{3/2}$)
- Fast multipole method (Greengard, 1987): ~O(N)
- Particle mesh Ewald (Darden, 1993): ~O(N*log*N)

How do you practically run a MD simulation?

8. Numerical temperature control (thermostats)

□ Basic MD

- NVE (micro-canonical ensemble): an adiabatic process with no heat exchange. Total energy of the system is conserved.

□ Other common ensembles

- **NVT** (canonical ensemble): constant temperature MD. In NVT, the energy of endothermic and exothermic processes is exchanged with a thermostat.
- NPT (isothermal-isobaric ensemble): In addition to a thermostat, a barostat is needed. It corresponds most closely to laboratory conditions with a flask open to ambient temperature and pressure.
- μVT (grand-canonical ensemble): chemical potential, temperature and volume are constants. The system exchange energy and particles with a reservoir, so that various possible states of the system can differ in both their total energy and total number of particles.

How do you practically run a MD simulation?

8. Numerical temperature control (thermostats)

1. Andersen thermostat

The simplest thermostat which does correctly sample the NVT ensemble

At each step, some prescribed number of particles is selected, and their velocities are drawn from a Gaussian distribution at the prescribed temperature

$$P(v) = \left(\frac{\beta}{2\pi m}\right)^{3/2} \exp\left[-\beta m v^2/2\right]$$

The strength of the coupling to the heat bath is specified by a collision frequency, ν. For each particle, a random variate is selected between 0 and 1. If this variate is less than $\nu\Delta t$, then that particle's velocities are reset.

2. Langevin thermostat

At each time step, all particles receive a **friction force** and a **random force** that satisfy the fluctuation-dissipation theorem, thereby guaranteeing NVT statistics.

$$m_i \frac{d^2 r_i}{dt^2} = -\nabla_i V - \gamma v_i + \boldsymbol{W}_i(t)$$

where γ is a friction coefficient, and $\boldsymbol{W}_i(t)$ is a random force with

$$\langle W_i(t) W_j(t') \rangle = \delta_{ij}\delta(t-t')6\gamma k_B T$$

How do you practically run a MD simulation?

8. Numerical temperature control (thermostats)

3. Nose-Hoover thermostat

A Hamiltonian with an extra degree of freedom for heat bath, s, is introduced

$$H_{Nose} = \sum_{i=1}^{N} \frac{p_i^2}{2m_i s^2} + V(q_i) + \frac{p_s^2}{2Q} + gk_B T ln(s)$$

According to the Hamiltonian formalism, we define the equations of motion by using the extended Hamiltonian

$$\frac{d\mathbf{q}_i}{dt} = \frac{\partial H_{\text{Nose}}}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i s^2}$$

$$\frac{d\mathbf{p}_i}{dt} = -\frac{\partial H_{\text{Nose}}}{\partial \mathbf{q}_i} = -\frac{\partial \phi}{\partial \mathbf{q}_i}$$

$$\frac{ds}{dt} = \frac{\partial H_{\text{Nose}}}{\partial p_s} = \frac{p_s}{Q}$$

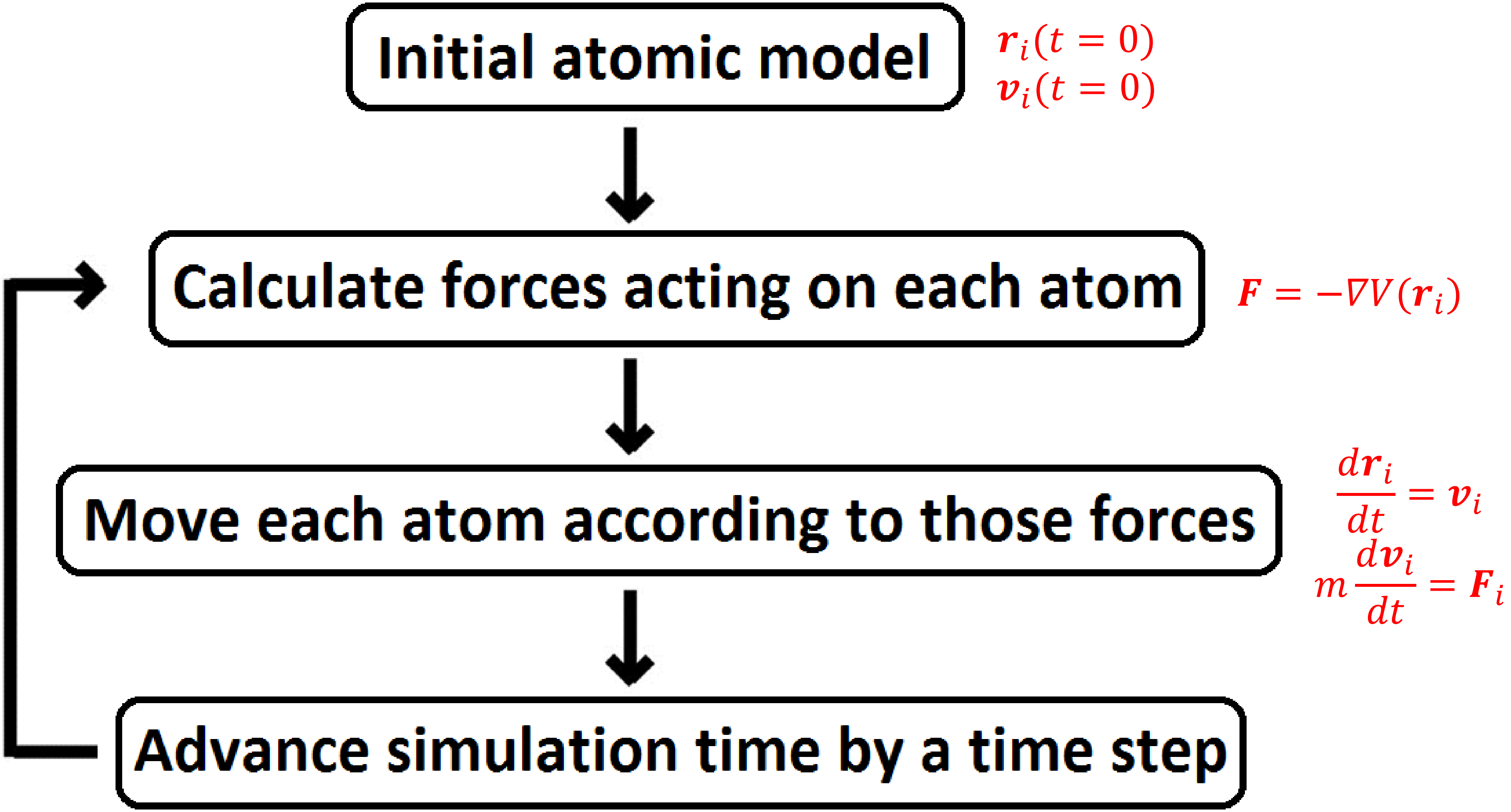
$$\frac{dp_s}{dt} = -\frac{\partial H_{\text{Nose}}}{\partial s} = \frac{\sum \frac{\mathbf{p}_i^2}{m_i s^2} - gkT}{s}$$

It can be approved that the partition function of the extended system is equivalent to that of the original system in the canonical ensemble except for a constant factor.

4. Dissipative particle dynamics thermostat **(will be introduced in Part II.)**

How do you practically run a MD simulation?

Procedure of MD simulation



Initial atomic model

$$\boldsymbol{r}_i(t=0)$$
$$\boldsymbol{v}_i(t=0)$$

Calculate forces acting on each atom

$$\boldsymbol{F} = -\nabla V(\boldsymbol{r}_i)$$

Move each atom according to those forces

$$\frac{d\boldsymbol{r}_i}{dt} = \boldsymbol{v}_i$$
$$m\frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{F}_i$$

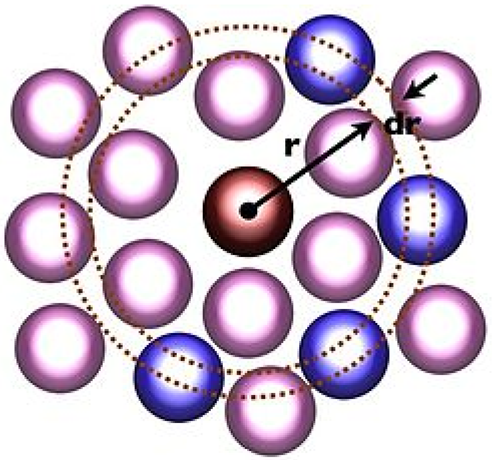
Advance simulation time by a time step

Part I: Molecular Dynamics

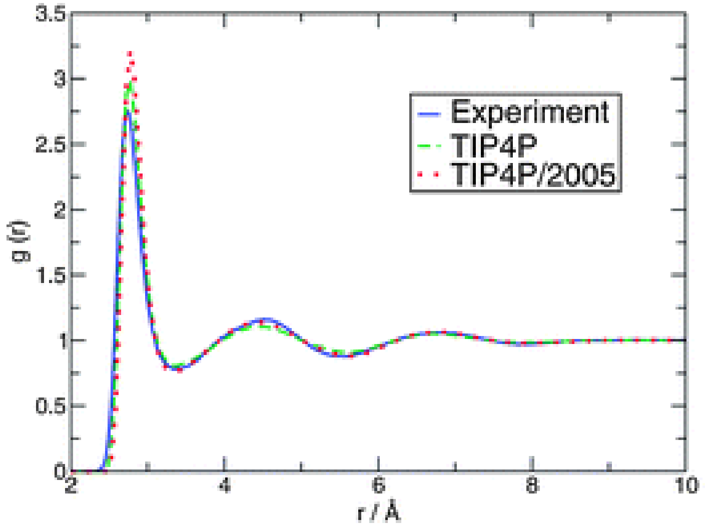
- Basics of MD simulation
  - motivation, history, typical length and time scales
- Potentials
  - Non-bonded interactions
  - Bonded interactions
- Algorithms for time integration
  - Verlet, velocity-Verlet, Beeman
- How do you practically run a MD simulation
  - Scaling, Periodic BC, Potential cut-offs, cell-list and Verlet list, thermostats
- **Analysis of MD**
  - Configurations, time correlations, transport properties

Analysis of MD

☞ static properties such as structure, energy, and pressure are obtained from **pair (radial) distribution functions**



$$\boldsymbol{g}(\boldsymbol{r}) = \frac{\boldsymbol{n}(\boldsymbol{r})}{4\pi\, \boldsymbol{r}^2\, \Delta \boldsymbol{r}} \frac{1}{\rho}$$



Oxygen–oxygen radial distribution function for liquid water at *T* = 298 K (Vega, et. al, Faraday Discuss. 2009.)

- *g(**r**)dr* is the probability of finding a particle in volume $d^3\mathbf{r}$ around **r** given one at **r** =0
- *g(r)* -> 0 as r -> 0 due to the strong repulsive forces
- *g(r)* tends to 1 as r at large distances, no long-range order
- g(r)=1 for Idea gas

Analysis of MD

☞ other outputs

1. Kinetic Energy

$$\bar{K} = \lim_{t\to\infty} \frac{1}{t} \int_{t_0}^{t_0+t} \sum_i \frac{1}{2} m_i v_i^2(\tau)\, d\tau$$

According to the equipartition theorem, we have

$$\langle \frac{1}{2} m_i v_i^2 \rangle = \frac{d}{2} N k_B T$$ where d is the dimensionality

This defines the temperature of MD system.

2. Potential Energy

$$\bar{V} = \lim_{t\to\infty} \frac{1}{t} \int_{t_0}^{t_0+t} \sum_{i<j} V(r_i(\tau) - r_j(\tau)\, d\tau = \frac{N\rho}{2} \int V(r) g(r) d^3 r$$

3. Pressure

$$\mathrm{P} = \rho \mathrm{k_B T} + \frac{2\pi\rho^2}{3} \int_0^\infty dr r^3 f(r) g(r) \qquad \mathrm{P} = \rho \mathrm{k_B T} + \frac{1}{3V} \left\langle \sum_i \boldsymbol{r}_i \cdot \boldsymbol{F}_i \right\rangle$$

Analysis of MD

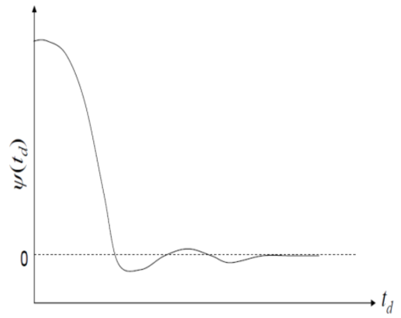
☞ dynamic and transport properties are obtained from time correlation functions

Velocity autocorrelation function

$$\psi(t_d) = \frac{1}{N}\left\langle \sum_{i}^{N} \vec{v}_i(t_0)\cdot\vec{v}_i(t_0+t_d) \right\rangle$$

Kinetic energy $\psi(0) = \dfrac{3kT}{m}$

Diffusivity $D = \int_0^\infty \psi(t_d)$



Stress autocorrelation function

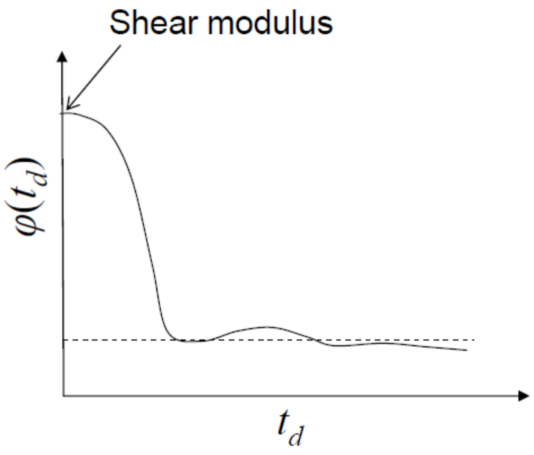
$$J = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{bmatrix}$$

$$\varphi(t_d) = \frac{\rho}{3kT}\frac{1}{N}\sum \left\langle J_{\alpha\beta}(t_0) J_{\alpha\beta}(t_0+t_d) \right\rangle$$

$$J_{\alpha\beta} = m\sum_{i}^{N} v_{i\alpha} v_{i\beta} + \frac{1}{2}\sum_{i\neq j}^{N} r_{ij\beta} F_{ij\alpha}$$

Shear modulus $\varphi(0)$

Viscosity $\eta = \int_0^\infty \varphi(t_d)$



Analysis of MD

☞transport properties

Example: 1D diffusion

$$\frac{\partial N}{\partial t} = D\frac{\partial^2 N}{\partial x^2} \qquad N(x,t) = \frac{N_0}{2\sqrt{\pi Dt}}\exp\left[\frac{-x^2}{4Dt}\right]$$

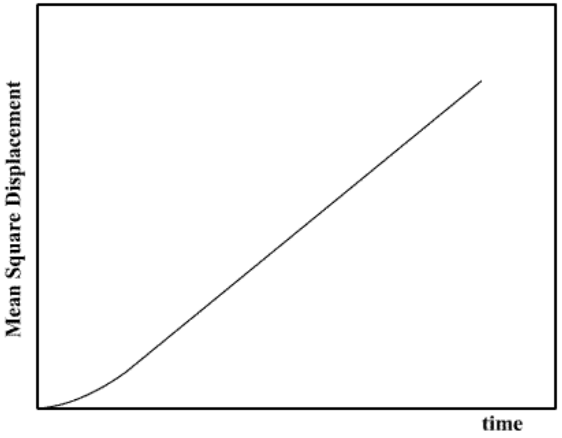
Second moment of the distribution is the mean-square displacement

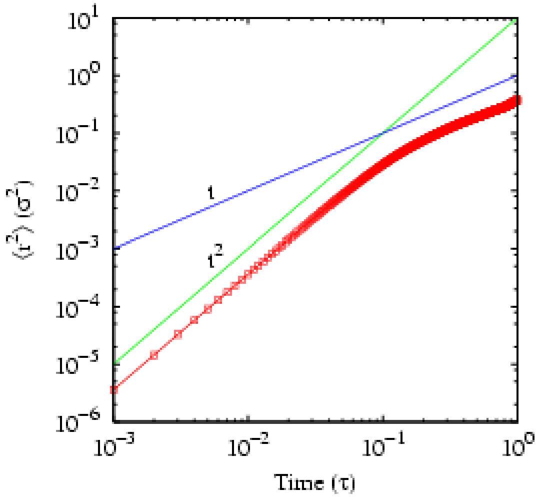
$$\left\langle [x(t)-x(0)]^2 \right\rangle = \frac{1}{N_0}\int x^2 N(x,t)dx$$

Einstein’s relation

$$\left\langle [x(t)-x(0)]^2 \right\rangle = 2Dt$$

Applicable when the time is large compared to the average time between atomic collisions





At short time

$x(t) = vt$

$[x(t)]^2 = [vt]^2 \sim t^2$

Analysis of MD

☞transport properties

Example: 1D diffusion

$$\dot{x}(t) = \frac{dx}{dt} \qquad\qquad x(t) - x(0) = \int_0^t \dot{x}(t')dt'$$

Square both sides and average over time origins

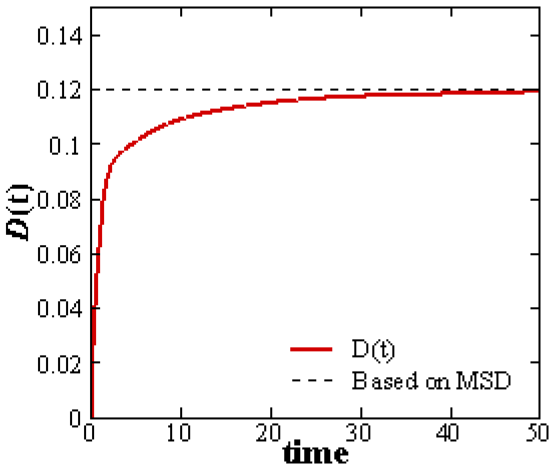
$$msd = \left\langle \left[x(t) - x(0)\right]^2 \right\rangle = \int_0^t dt'' \int_0^t dt' \left\langle \dot{x}(t')\dot{x}(t'') \right\rangle$$

Use integrand symmetry, shift the time origin, then we obtain

$$\frac{\left\langle \left[x(t) - x(0)\right]^2 \right\rangle}{2t} = \int_0^t d\tau \left\langle \dot{x}(\tau)\dot{x}(0) \right\rangle \left(1 - \frac{\tau}{t}\right)$$

Take the long-time limit

$$\lim_{t\to\infty} \frac{\left\langle \left[x(t) - x(0)\right]^2 \right\rangle}{2t} = \int_0^\infty d\tau \left\langle \dot{x}(\tau)\dot{x}(0) \right\rangle$$

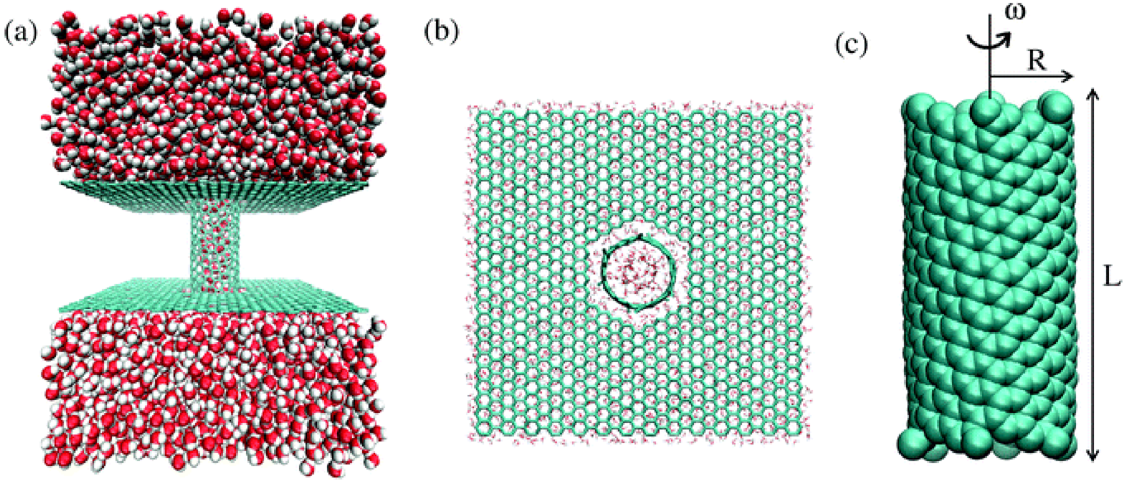


Consider Einstein's relation $D = \int_0^\infty d\tau \left\langle \dot{x}(\tau)\dot{x}(0) \right\rangle$

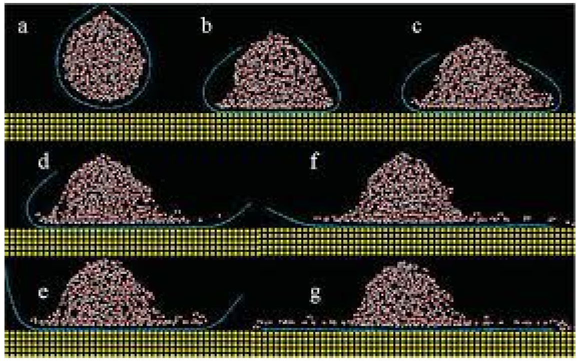
$$D = \frac{1}{3}\int_0^\infty \langle \boldsymbol{v}(t)\boldsymbol{v}(0)\rangle dt$$

Green-Kubo relation (3D)

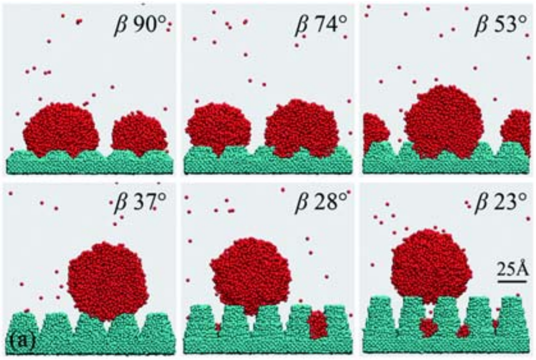
Examples of MD simulation of simple fluids



Pumping of water by rotating chiral carbon nanotube. Feng, et. al., Nanoscale, 2014.



Precursor Film in Dynamic Wetting, Yuan and Zhao, PRL, 2010.



Effect of nano structures on the nucleus wetting modes, Xu, et al., RSC Adv. 2016.

Software/Package for MD simulation

LAMMPS: http://lammps.sandia.gov/

GROMACS FAST. FLEXIBLE. FREE. http://www.gromacs.org/

NAMD Scalable Molecular Dynamics http://www.ks.uiuc.edu/Research/namd/

HOOMD-blue: http://codeblue.umich.edu/hoomd-blue/

AMBER

CHARMM

Materials Studio

RedMD

ESPResSo … …

Part II: Dissipative Particle Dynamics

- **Brief introduction of DPD**
  - motivation, history
- **Foundations of DPD**
  - Coarse-graining of MD (Mori-Zwanzig formalism)
- **Parameterization of a DPD system**
  - Force field of classic DPD
  - Pressure, compressibility, viscosity, diffusivity of DPD fluids
  - Scaling
  - Modeling of complex fluids using DPD
- **How do you practically run a DPD simulation**
  - Implementation of boundary conditions
  - Computing viscosity, diffusivity
- **Some Applications**
  - Droplet, Blood flows, Self-assembly Dynamics

Why CG/mesoscopic?

MESOscale: A Scale Between Them

MACROscale
continuum description

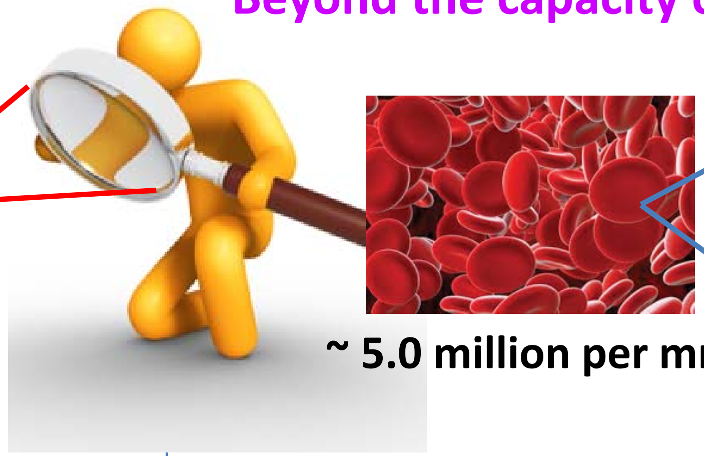
Navier-Stokes equation

Continuum fluid mechanics

Mesh: FDM, FEM, FVM …

Mesh-free: SPH

Discontinuum Nature
Thermal Fluctuation
Beyond the capacity of MD



~ 5.0 million per mm^3

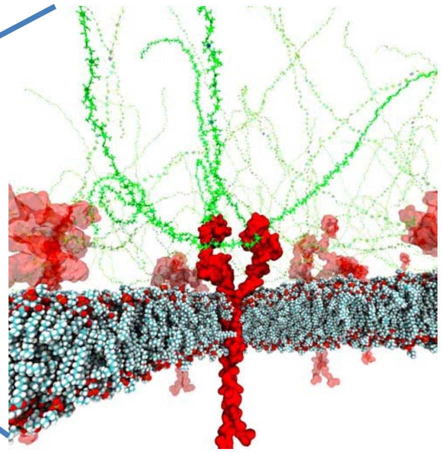
Coarse-graining of molecules
Loses dynamical details of atom/molecules
Considers thermal fluctuation

Mesh: LBM, FH

Mesh-free: CGMD, DPD, SDPD

MICROscale
Atomistic description

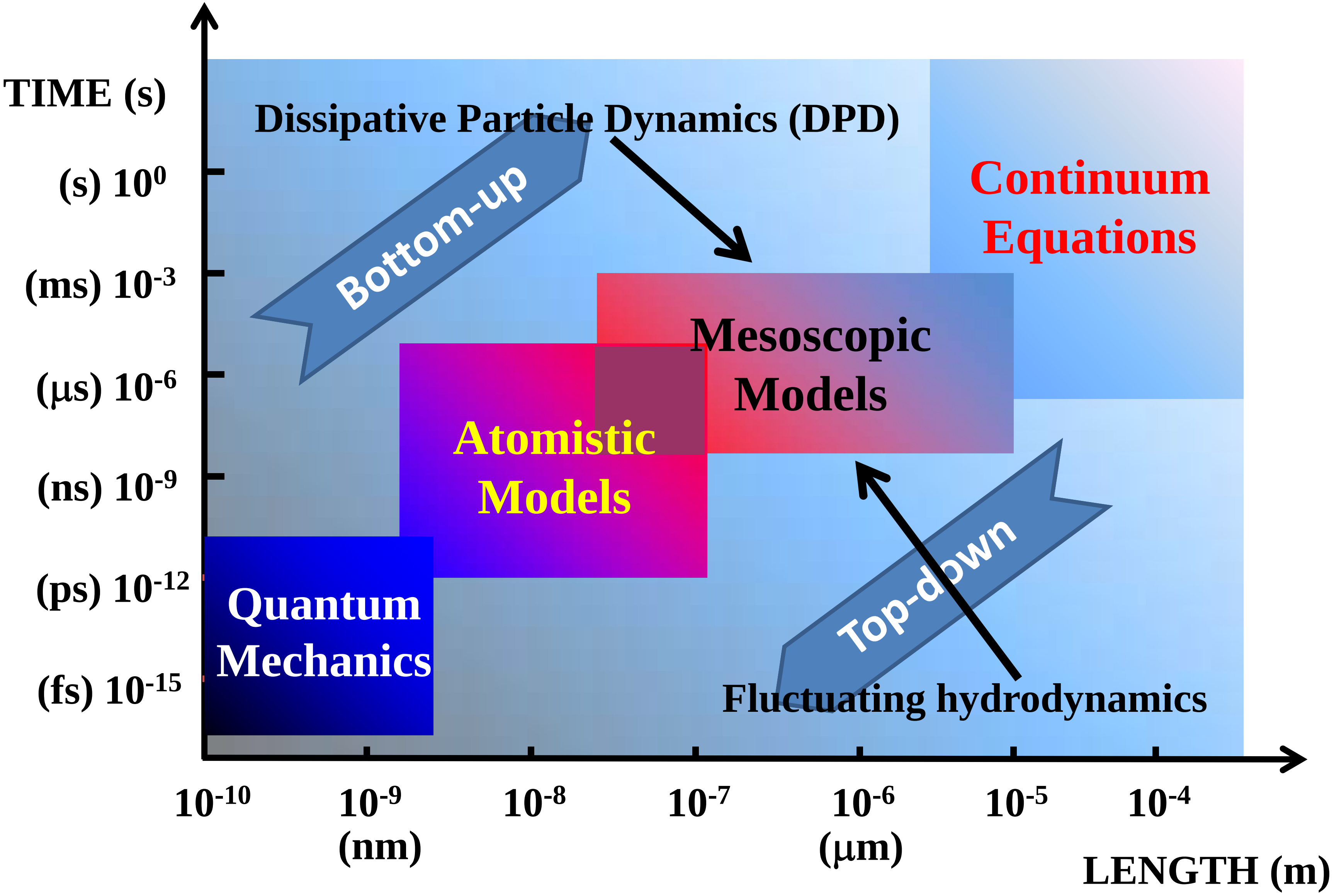
Molecular dynamics



Atomistic description
Contains details of atom/molecules
Limited in time/length

Mesh-free: MD

Length and time scales for different models

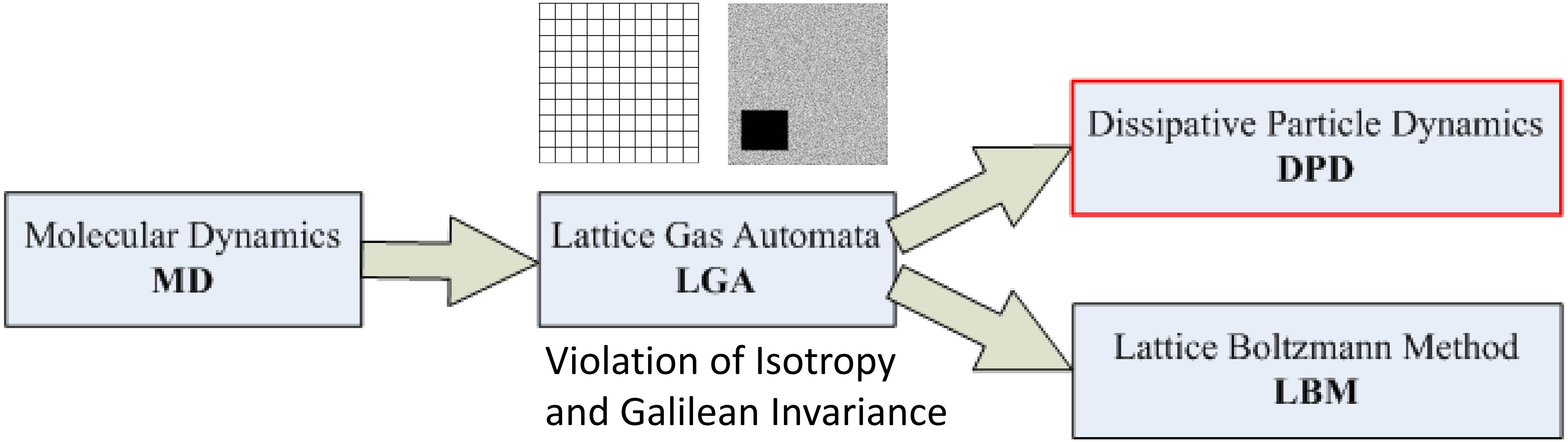


D. G. Vlachos, Adv. Chem. Eng., 2005, **30**: 1-61.

History of DPD method

❖Babyhood (1992-1995)

- ➢ Original formulation (Hoogerbrugge & Koelman, 1992)



Molecular Dynamics
MD

Lattice Gas Automata
LGA

Dissipative Particle Dynamics
DPD

Lattice Boltzmann Method
LBM

Violation of Isotropy
and Galilean Invariance

❖Youth (1995-2003)

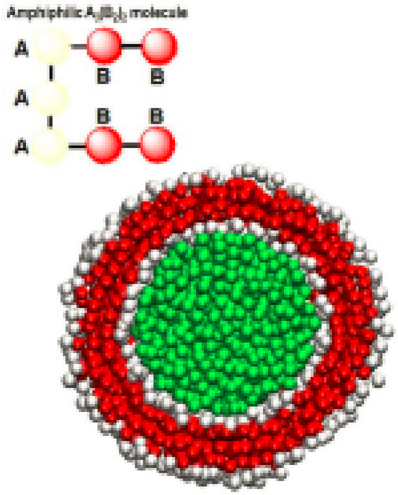
- ➢ Fluctuation-dissipation relation (Español & Warren, 1995)
- ➢ Important contributions to the DPD methodology (model of polymers, implementation of boundary conditions, et al.)

❖ Golden Era (2003-now)

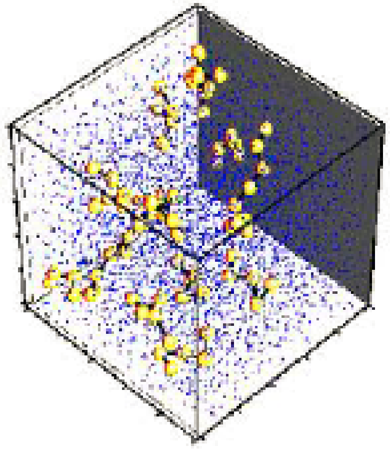
- ➢ Successful applications to material science, biological and biomedical systems, fluid rheology and other complex fluids.

Brief Introduction of DPD

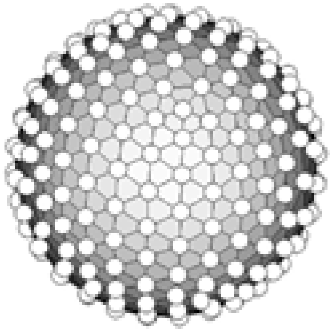
Successful DPD applications



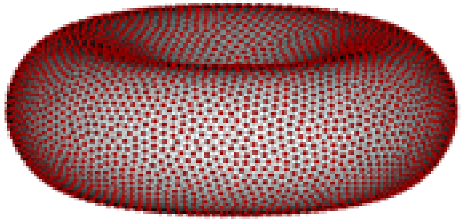
Polymer self-assembly



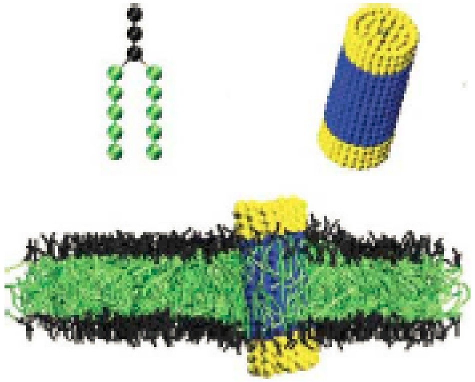
DNA



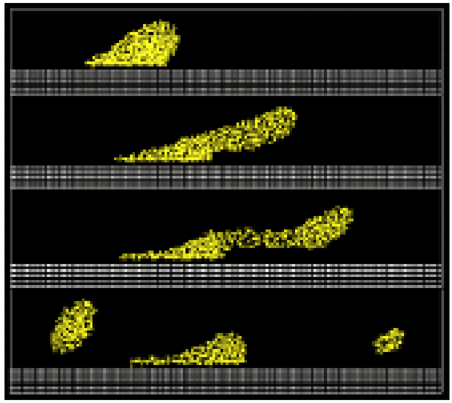
Colloids



Blood

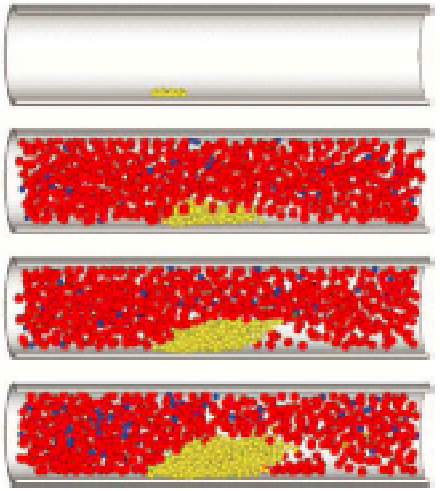


Membrane



Droplet

Surfactant



Platelets

Brief Introduction of DPD

Governing equation of DPD

- Particles in DPD represent clusters of molecules and interact through simple pair-wise forces

$$\boldsymbol{F}_i = \sum_{i \neq j} \left(\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R\right) \qquad \begin{aligned} \boldsymbol{F}_{ij}^C &= \alpha\left(1 - r_{ij}/r_c\right)\boldsymbol{e}_{ij} \\ \boldsymbol{F}_{ij}^D &= \gamma\left(1 - r_{ij}/r_c\right)^2\left(\boldsymbol{e}_{ij}\boldsymbol{v}_{ij}\right)\boldsymbol{e}_{ij} \\ \boldsymbol{F}_{ij}^R &= \sigma\left(1 - r_{ij}/r_c\right)dt^{-1/2}\xi_{ij}\boldsymbol{e}_{ij} \end{aligned}$$

- DPD system is thermally equilibrated through a thermostat defined by forces $\boldsymbol{F}_{ij}^D$ and $\boldsymbol{F}_{ij}^R$ via $\sigma^2 = 2\gamma k_B T$.
- The time evolution equations are given by:

$$d\mathbf{r}_i = \mathbf{v}_i dt, \qquad d\mathbf{v}_i = \mathbf{F}_i dt$$

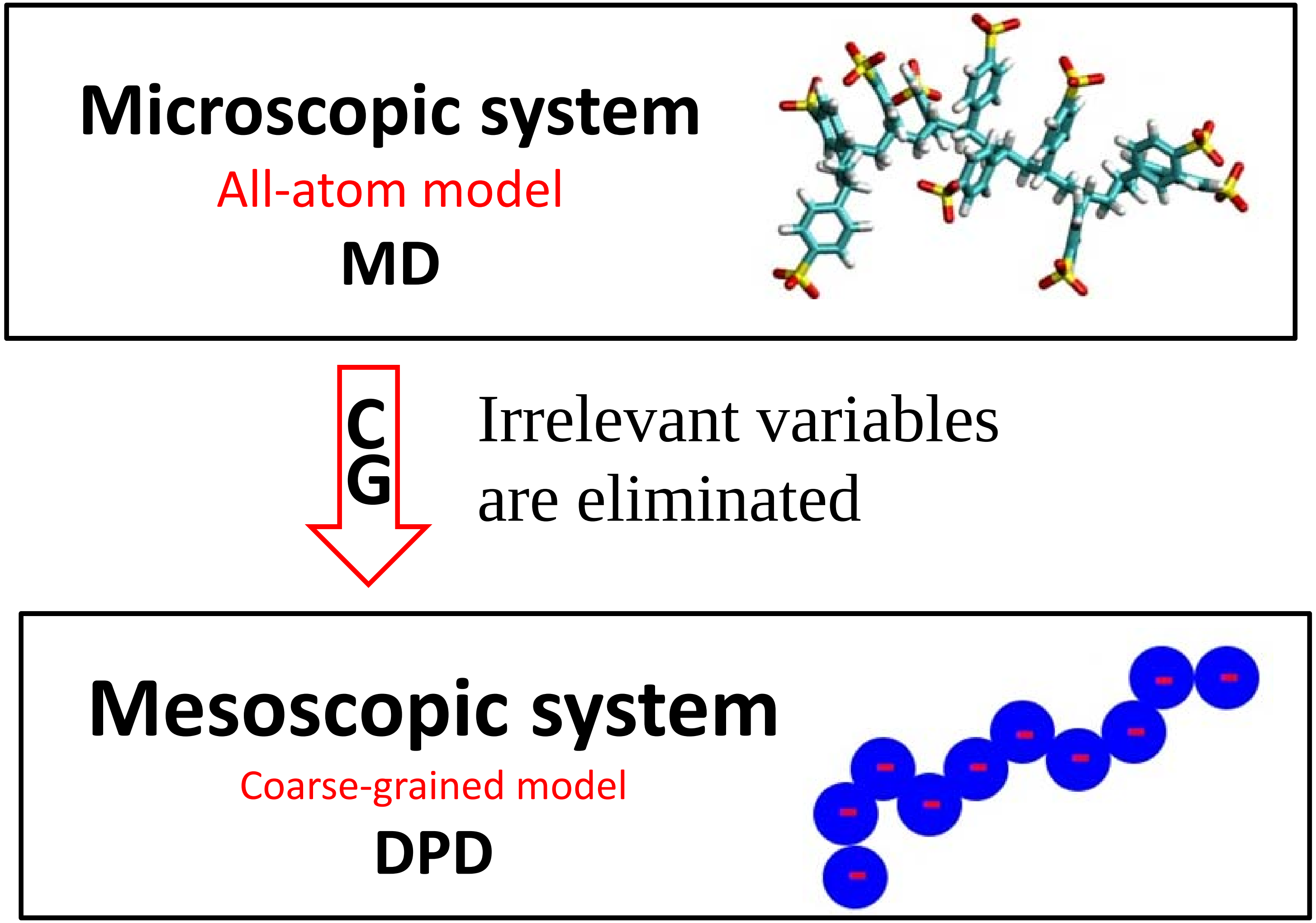
Hoogerbrugge & Koelman, EPL., 1992

Part II: Dissipative Particle Dynamics

- Brief introduction of DPD
  - motivation, history
- **Foundations of DPD**
  - Coarse-graining of MD (Mori-Zwanzig formalism)
- Parameterization of a DPD system
  - Force field of classic DPD
  - RDF, pressure, compressibility, viscosity, diffusivity of DPD fluids
  - Scaling
  - Modeling of complex fluids using DPD
- How do you practically run a DPD simulation
  - Implementation of boundary conditions
  - Computing viscosity, diffusivity
- Some Applications
  - Droplet, Blood flows, Self-assembly Dynamics

Where does DPD come from?

DPD is a bottom-up particle-based mesoscopic method from coarse-graining of MD system.



Microscopic system

All-atom model

MD

CG

Irrelevant variables are eliminated

Mesoscopic system

Coarse-grained model

DPD

Elimination of degrees of freedom from a system

Consider a linear differential system for two variables:

$$\frac{dx}{dt} = x + y, \qquad (1)$$

$$\frac{dy}{dt} = -y + x, \qquad (2)$$

Let $x_0 = x(t = 0)$ and $y_0 = y(t = 0)$ denote the corresponding initial values. By solving the Eq. (2)

$$y = \int_0^t e^{-(t-s)} x(s) ds + y_0 e^{-t}$$

we can reduce the system into an equation for $x(t)$ alone:

$$\frac{dx}{dt} = x + \int_0^t e^{-(t-s)} x(s) ds + y_0 e^{-t}$$

The second term in above equation introduces memory.

Dimension Reduction leads to memory effect and noise term.

Mori-Zwanzig Formalism (Zwanzig, Nonequilibrium Statistical Mechanics, 2001)

$$\frac{d\phi_i(x,t)}{dt} = R(\phi(x,t)), \quad \phi_i(x,0) = x_i, \quad 1 \le i \le n.$$

$\phi = (\hat{\phi}, \tilde{\phi}), \quad \hat{\phi} = (\phi_1, \ldots, \phi_m), \quad \tilde{\phi} = (\phi_{m+1}, \ldots, \phi_n)$

Form the Liouville equation $u_t = Lu$, the components $\hat{\phi}$ are

$$\hat{\phi}_j(x,t) = e^{tL} x_j$$

Let $\mathbb{P}$ be the conditional expectation projection $\mathbb{P}g(x) = E[g|\hat{x}]$.

Define $\mathbb{Q} = I - \mathbb{P}$ and keep in mind that $\mathbb{P}^2 = \mathbb{P}$, $\mathbb{Q}^2 = \mathbb{Q}$, and $\mathbb{P}\mathbb{Q} = 0$, as must be true for any projection.

$$\frac{\partial}{\partial t} e^{tL} x_j = L e^{tL} x_j = e^{tL} L x_j = e^{tL} \mathbb{P} L x_j + e^{tL} \mathbb{Q} L x_j$$

Dyson's formula $\quad e^{tL} = e^{t\mathbb{Q}L} + \int_0^t e^{(t-s)L} \mathbb{P} L e^{s\mathbb{Q}L} \, ds$

$$\boxed{\frac{\partial}{\partial t} e^{tL} x_j = e^{tL} \mathbb{P} L x_j + \int_0^t e^{(t-s)L} \mathbb{P} L e^{s\mathbb{Q}L} \mathbb{Q} L x_j \, ds + e^{t\mathbb{Q}L} \mathbb{Q} L x_j}$$

This is the Mori-Zwanzig equation. This equation is exact and is an alternative way of writing the original system.

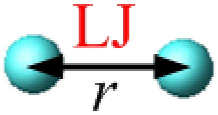
Mori-Zwanzig Formalism (Zwanzig, Nonequilibrium Statistical Mechanics, 2001)

Consider an atomistic system consisting of ***N*** atoms which are grouped into ***K*** clusters, and $\boldsymbol{N_C}$ atoms in each cluster.

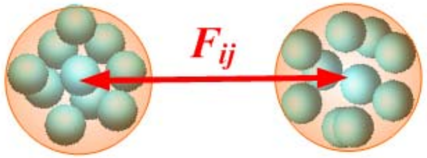
The Hamiltonian of the atomistic system is:

$$H=\sum_{\mu=1}^{K}\sum_{i=1}^{N_C}\frac{\mathbf{p}_{\mu,i}^2}{2m_{\mu,i}}+\frac{1}{2}\sum_{\mu,\nu}\sum_{i,j\neq i}V_{\mu i,\nu j}$$

Atomistic Model



Our interest is on the molecular or CG level :



$$M_\mu=\sum_{\mu,i}m_{\mu,i}$$

$$\mathbf{P}_\mu=\sum_{\mu,i}\mathbf{p}_{\mu,i}$$

$$\mathbf{R}_\mu=\frac{1}{M_\mu}\sum_{\mu,i}m_{\mu,i}\mathbf{r}_{\mu,i}$$

The equation of motion for CG particles can be written as:

$$\dot{\mathbf{P}}_I=k_BT\frac{\partial}{\partial\mathbf{R}_I}\ln\omega(\mathbf{R})$$

$$-\frac{1}{k_BT}\sum_{X=1}^{K}\int_0^t ds\left\langle[\delta\mathbf{F}_I(t-s)]\left[\delta\mathbf{F}_X(0)^T\right]\right\rangle\cdot\frac{\mathbf{P}_X(s)}{M_X}$$

$$+\delta\mathbf{F}_I(t)$$

Details see Z. Li et al, *Soft Matter*, **10**, 8659, 2014.

Bottom-up coarse-grained model:

The equation of motion (EOM) of the coarse-grained (CG) particles obtained from the Mori-Zwanzig projection is in a form of generalized Langevin equation, which is given by

$$\begin{aligned}\frac{d}{dt}\mathbf{P}_I &= \frac{1}{\beta}\frac{\partial}{\partial \mathbf{R}_I}\ln\omega(\mathbf{R}) \\ &- \beta\sum_{X=1}^{K}\int_0^t ds\,\left\langle[\delta\mathbf{F}_I(t-s)][\delta\mathbf{F}_X(0)]^T\right\rangle\frac{\mathbf{P}_X(s)}{M_X} \\ &+ \delta\mathbf{F}_I(t)\ ,\end{aligned}$$

First approximation: Here, we assume that the non-bonded interactions between neighboring clusters in the microscopic system are explicitly **pairwise decomposable**, and hence the total force consists of pairwise forces, e.g. $\mathbf{F}_I \approx \sum_{J\neq I}\mathbf{F}_{IJ}$ and $\delta\mathbf{F}_I \approx \sum_{J\neq I}\delta\mathbf{F}_{IJ}$.

Second approximation: In practice, we **neglect the many-body correlations** between different pairs, and assume that the force $\mathbf{F}_{IJ}$ between two clusters I and J depends only on the relative COM positions $\mathbf{R}_I$ and $\mathbf{R}_J$ and is independent of the positions of the rest of clusters.

Evaluation of coarse-grained interactions:

First term: Conservative Force:

$$\frac{1}{\beta}\frac{\partial}{\partial \mathbf{R}_I}\ln\omega(\mathbf{R}) = \langle \mathbf{F}_I \rangle \approx \sum_{J\neq I}\langle \mathbf{F}_{IJ}\rangle = \sum_{J\neq I} F^C_{IJ}(R_{IJ})\mathbf{e}_{IJ}$$

Second term: Dissipative Force:

$$-\quad \beta\sum_{X=1}^{K}\int_0^t ds\,\left\langle [\delta\mathbf{F}_I(t-s)][\delta\mathbf{F}_X(0)]^T\right\rangle \frac{\mathbf{P}_X(s)}{M_X}$$

Based on the second approximation,
the correlation of fluctuating forces between different pairs is ignored.
Thus, we have

$$\begin{aligned}
&\left\langle [\delta\mathbf{F}_I(t-s)][\delta\mathbf{F}_X(0)]^T\right\rangle \frac{\mathbf{P}_X(s)}{M_X} \\
&= \sum_{J\neq I}\sum_{Y\neq X}\left\langle [\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{XY}(0)]^T\right\rangle \mathbf{V}_X(s) \\
&= \sum_{J\neq I}\left\langle [\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{IJ}(0)]^T\right\rangle \mathbf{V}_I(s)|_{X=I,Y=J} + \\
&\qquad \left\langle [\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{JI}(0)]^T\right\rangle \mathbf{V}_J(s)|_{X=J,Y=I} \\
&= \sum_{J\neq I}\left\langle [\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{IJ}(0)]^T\right\rangle \mathbf{V}_{IJ}(s)
\end{aligned}$$

Markovian approximation:

Remark: The memory term given by Eq. (8) can be further simplified with a Markovian assumption that the memory of fluctuating force in time is short enough to be approximated by a Dirac delta function

$$\beta\langle[\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{IJ}(0)]^T\rangle = 2\gamma_{IJ}\delta(t-s)\ ,$$
$$\beta \int_0^t ds\, \left\langle[\delta\mathbf{F}_{IJ}(t-s)][\delta\mathbf{F}_{IJ}(0)]^T\right\rangle \mathbf{V}_{IJ}(s) = \gamma_{IJ}\cdot\mathbf{V}_{IJ}(t)\ ,$$

where γ_{IJ} is the friction tensor defined by $\gamma_{IJ} = \beta\int_0^\infty dt\,\langle[\delta\mathbf{F}_{IJ}(t)][\delta\mathbf{F}_{IJ}(0)]^T\rangle$. Then, the equation of motion of DPD particles based on the Markovian approximation can be expressed by

$$\frac{d\mathbf{P}_I}{dt} = \sum_{J\neq I}\{\, F^C_{IJ}(R_{IJ})\mathbf{e}_{IJ} - \gamma_{IJ}(R_{IJ})\,(\mathbf{e}_{IJ}\cdot\mathbf{V}_{IJ})\,\mathbf{e}_{IJ} + \delta\mathbf{F}_{IJ}(t)\,\}$$

DPD model

Where does DPD come from?

Answer: DPD comes from coarse-graining of its underlying microscopic system.

- Irrelevant variables are eliminated using MZ projection.
- Only resolve the variables that we are interested in.
- Unresolved details are represented by the dissipative and random forces.

Part II: Dissipative Particle Dynamics

- Brief introduction of DPD
  - motivation, history
- Foundations of DPD
  - Coarse-graining of MD (Mori-Zwanzig formalism)
- **Parameterization of a DPD system**
  - Force field of classic DPD
  - Pressure, compressibility, viscosity, diffusivity of DPD fluids
  - Scaling
  - Modeling of complex fluids using DPD
- How do you practically run a DPD simulation
  - Implementation of boundary conditions
  - Computing viscosity, diffusivity
- Some Applications
  - Droplet, Blood flows, Self-assembly Dynamics

Parameterization of a DPD system

Force field of classic DPD

$$\boldsymbol{F}_i = \sum_{i \neq j} \left(\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R\right)$$

$$\boldsymbol{F}_{ij}^C = a\left(1 - r_{ij}/r_c\right)\boldsymbol{e}_{ij}$$

$$\boldsymbol{F}_{ij}^D = \gamma\left(1 - r_{ij}/r_c\right)^2\left(\boldsymbol{e}_{ij}\boldsymbol{v}_{ij}\right)\boldsymbol{e}_{ij}$$

$$\boldsymbol{F}_{ij}^R = \sqrt{2\gamma k_B T}\left(1 - r_{ij}/r_c\right)dt^{-1/2}\xi_{ij}\boldsymbol{e}_{ij}$$

The conservative force $\boldsymbol{F}_{ij}^C$ is responsible for the static properties, i.e.,

Pressure

Compressibility

Radial distribution function g(r)

The dissipative force $\boldsymbol{F}_{ij}^D$ **and random force** $\boldsymbol{F}_{ij}^R$ together act as a thermostat and determine the dynamics properties, i.e.,

Viscosity

Diffusivity

Time correlation functions

Parameterization of a DPD system

DPD thermostat

$$\boldsymbol{F}_i = \sum_{i \neq j} \left(\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R \right)$$

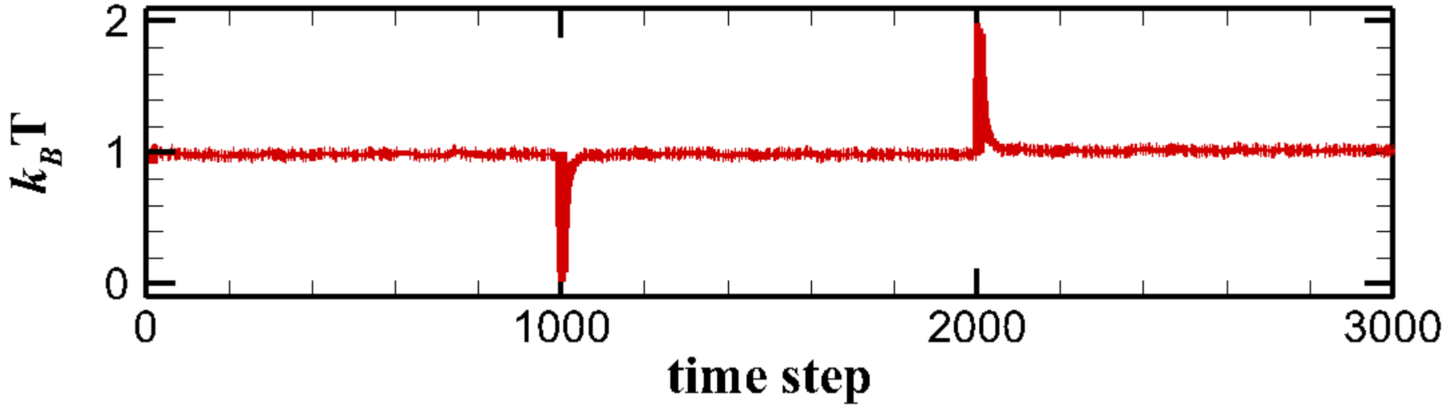
$$\boldsymbol{F}_{ij}^D = \gamma w_D(r_{ij})\left(\boldsymbol{e}_{ij}\boldsymbol{v}_{ij}\right)\boldsymbol{e}_{ij}$$

$$\boldsymbol{F}_{ij}^R = \sigma w_R(r_{ij}) dt^{-1/2} \xi_{ij}\boldsymbol{e}_{ij}$$

To satisfy the fluctuation-dissipation theorem (FDT):

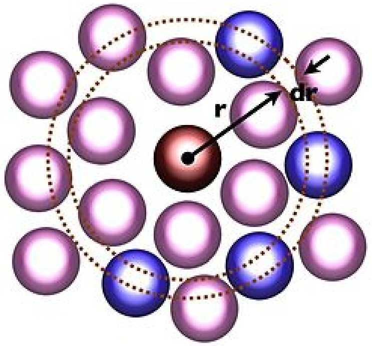
$$[w_R(r)]^2 = w_D(r) \text{ and } \sigma^2 = 2\gamma k_B T$$

Then, **the dissipative force $\boldsymbol{F}_{ij}^D$ and random force $\boldsymbol{F}_{ij}^R$** together act as a DPD thermostat.

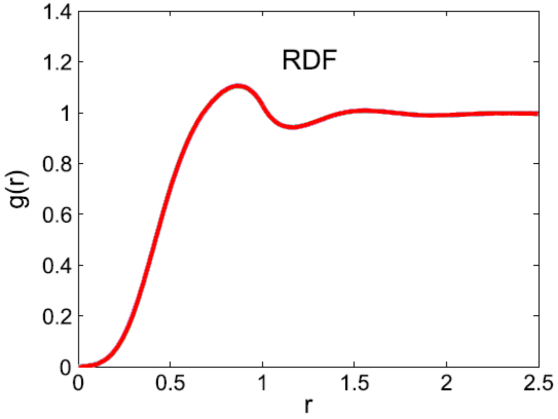


RDF, pressure, compressibility, viscosity, diffusivity of DPD fluids

Radial distribution function



$$\boldsymbol{g}(\boldsymbol{r}) = \frac{\boldsymbol{n}(\boldsymbol{r})}{4\pi\, \boldsymbol{r}^2\, \Delta \boldsymbol{r}}\frac{1}{\rho}$$



Pressure

$$\mathrm{P} = \rho \mathrm{k_B T} + \frac{1}{3V}\left\langle \sum_i \boldsymbol{r}_i \cdot \boldsymbol{F}_i \right\rangle$$

$$\mathrm{P} = \rho \mathrm{k_B T} + \frac{2\pi\rho^2}{3}\int_0^\infty r^3 f(r) g(r)\, dr$$

Compressibility

$$\kappa^{-1} = \frac{1}{k_B T}\left(\frac{\partial P}{\partial \rho}\right)_T$$

For linear conservative force

$$\boldsymbol{F}_{ij}^C = a\left(1 - r_{ij}/r_c\right)\boldsymbol{e}_{ij}$$

The equation of state is

$$P = \rho k_B T + 0.1 a \rho^2$$

Then

$$\kappa^{-1} = 1 + 0.2 a\rho / k_B T$$

RDF, pressure, compressibility, viscosity, diffusivity of DPD fluids

Diffusivity

Consider the motion of single particle given by Langevin equation

$$m\frac{dv}{dt} = -\frac{v_i}{\tau} + F_i^R$$

$$\frac{1}{\tau} = \sum_{j \neq i} \gamma w_D(r) \frac{\boldsymbol{e}_{ij}\boldsymbol{e}_{ij}}{3} = \frac{4\pi\gamma\rho}{3}\int_0^\infty r^2 w_D(r) g(r) dr$$

Self-diffusion coefficient

$$D = \frac{1}{3}\int_0^\infty \langle \boldsymbol{v}(t)\boldsymbol{v}(0)\rangle dt = \tau k_B T$$

Viscosity

There are two contributions to the pressure tensor: the *kinetic* part ν_K and the *dissipative* part ν_D

$$\nu_K = \frac{D}{2}$$

$$\nu_D = \frac{2\pi\gamma\rho}{15}\int_0^\infty r^4 w_D(r) g(r) dr$$

If $w_D(r) = (1 - r/r_c)^2$, and using $g(r) = 1$, we have

$$\nu = \frac{45 k_B T}{4\pi\gamma\rho r_c^3} + \frac{2\pi\gamma\rho r_c^5}{1575}$$

Groot, R.D. and P.B. Warren, *Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation.* J. Chem. Phys., 1997.
Marsh, C.A., G. Backx, and M.H. Ernst, *Static and dynamic properties of dissipative particle dynamics.* PRE, 1997.

Scaling

Basic units: Length, mass and time units

- The mass of the DPD particle is N_m times the mass of MD particle.

$$M = mN_m$$

- The cut-off radius can be found by equating mass densities of MD and DPD systems.

$$R_c = \left(\frac{N_m \rho^*_{\text{DPD}}}{\rho_{\text{MD}}}\right)^{\frac{1}{3}} \sigma$$

- The DPD conservative force coefficient *a* is found by equating the dimensionless compressibility of the systems.

$$a = k_B T \frac{\kappa^{-1} N_m - 1}{2\alpha\rho_{DPD}}$$

- The time scale is determined by insisting that the shear viscosities of the DPD and MD fluids are the same.

$$\tau_{\text{DPD}} = \frac{\nu^*_{\text{DPD}}}{\nu_{\text{MD}}} \left(\frac{R_c}{\sigma}\right)^2 \tau$$

- The variables marked with the symbol "*" have the same numerical values as in DPD but they have units of MD.

Groot & Warren, J. Chem. Phys., 1997

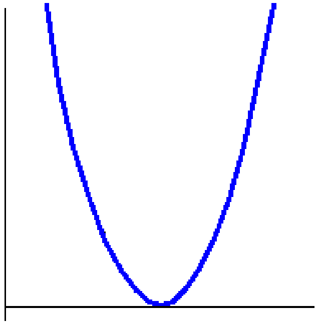
Keaveny, Pivkin, Maxey & Karniadakis, J. Chem. Phys., 2005

Modeling of complex fluids using DPD

Bonded interactions

Hookean spring (harmonic potential)

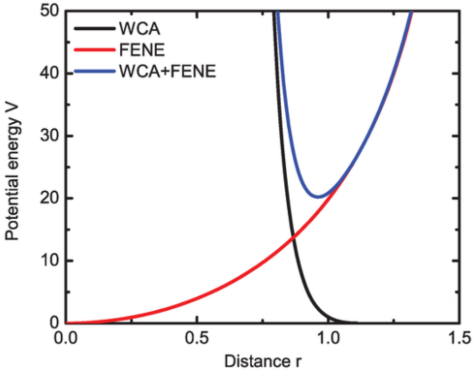
$$V = K_b(r - r_0)^2$$

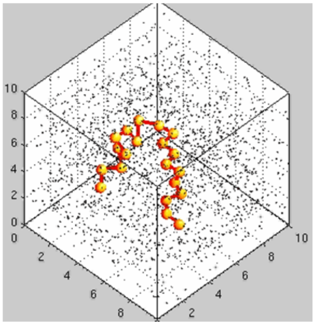


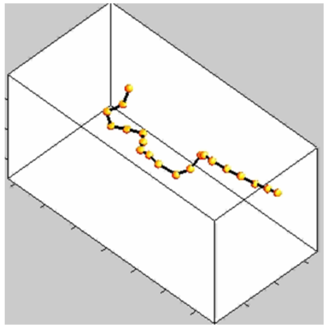
$$\mathrm{F} = -\frac{dV}{dr} = 2K_b(r_0 - r)$$

Finitely extensible nonlinear elastic (FENE) spring

$$\mathrm{V} = -0.5KR_0^2 \ln\left[1 - \left(\frac{r}{R_0}\right)^2\right] + 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] + \epsilon$$







Wormlike chain (WLC) (for semi-flexible polymers)

An interpolation formula that approximates the force-extension behavior is (Marko-Siggia formula)

$$F(r) = \frac{k_B T}{l_p}\left[\frac{1}{4}\left(1 - \frac{r}{L}\right)^{-2} + \frac{r}{L} - \frac{1}{4}\right]$$

L represents the contour length of the polymer chain.
l_p is persistence length, indicating the rigidity/flexibility of the polymer chain.

Symeonidis, V., G. Em Karniadakis, and B. Caswell, PRL, 2005.

Part II: Dissipative Particle Dynamics

- **Brief introduction of DPD**
  - motivation, history
- **Foundations of DPD**
  - Coarse-graining of MD (Mori-Zwanzig formalism)
- **Parameterization of a DPD system**
  - Force field of classic DPD
  - Compressibility, viscosity, diffusivity of DPD fluids
  - Scaling
  - Modeling of complex fluids using DPD
- **How do you practically run a DPD simulation**
  - Implementation of boundary conditions
  - Computing viscosity, diffusivity
- **Some Applications**
  - Droplet, Blood flows, Self-assembly Dynamics

Modified velocity-Verlet algorithm for DPD

DPD forces depend on velocity

$$\mathbf{r}_i(t+\Delta t)=\mathbf{r}_i(t)+\Delta t\,\mathbf{v}_i(t)+\tfrac{1}{2}(\Delta t)^2\mathbf{f}_i(t)$$

$$\tilde{\mathbf{v}}_i(t+\Delta t)=\mathbf{v}_i(t)+\boxed{\lambda}\Delta t\,\mathbf{f}_i(t)$$

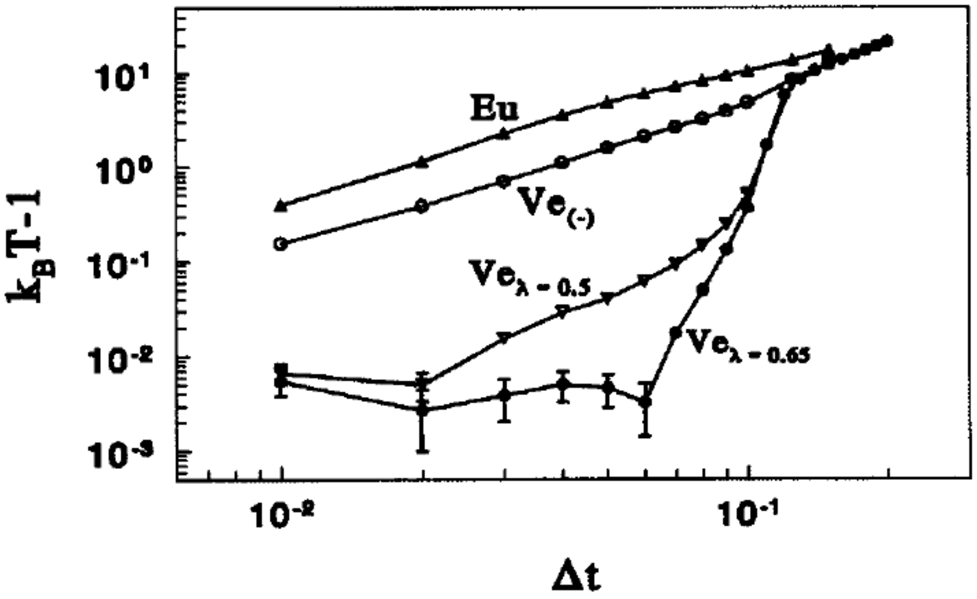
$$\mathbf{f}_i(t+\Delta t)=\mathbf{f}_i(\mathbf{r}(t+\Delta t),\tilde{\mathbf{v}}(t+\Delta t))$$

$$\mathbf{v}_i(t+\Delta t)=\mathbf{v}_i(t)+\tfrac{1}{2}\Delta t(\mathbf{f}_i(t)+\mathbf{f}_i(t+\Delta t))$$

Optimum value:

$$\lambda = 0.65$$

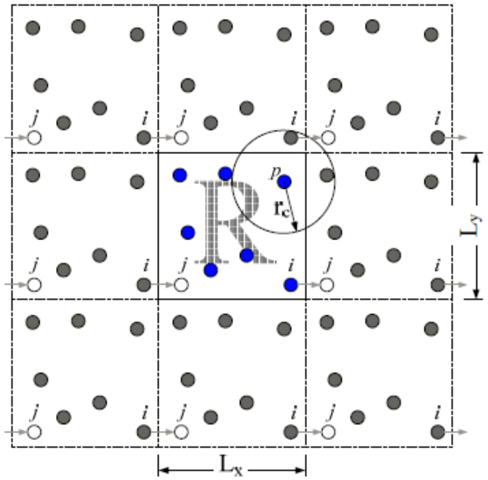
For this value the time step can be increased to 0.06 without significant loss of temperature control.



Groot, R.D. and P.B. Warren, J. Chem. Phys., 1997.

Implementation of Boundary conditions: PBC

Periodic boundary conditions

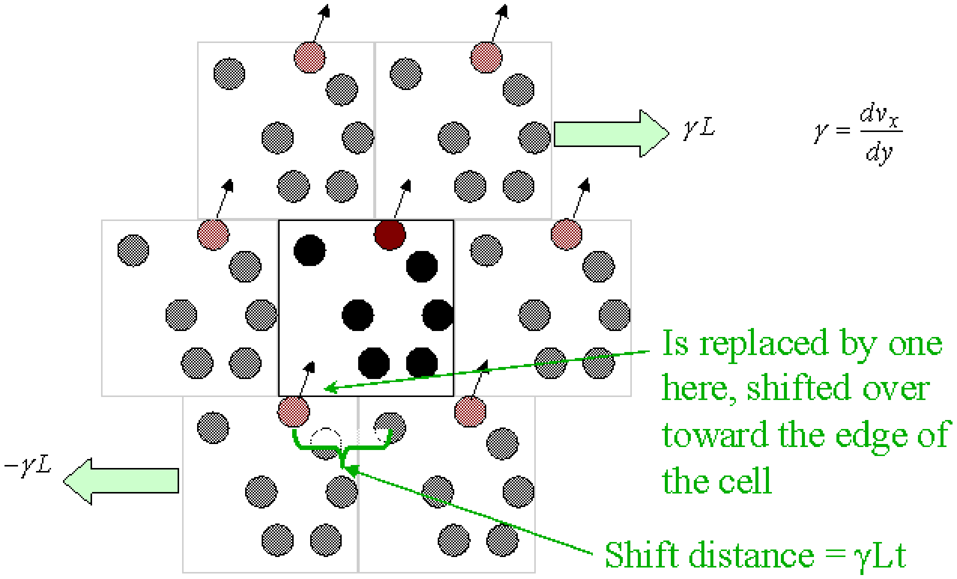


For computational box $0 < x < L$

If $x < 0$ then $x = x + L$
If $x > L$ then $x = x - L$

Minimum image criterion: among all images of a particle, consider only the closest and neglect the rest.

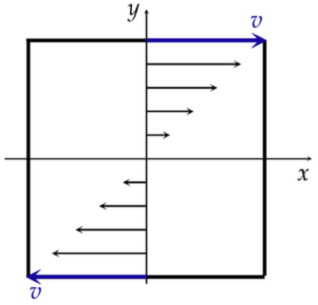
Lees-Edwards boundary conditions

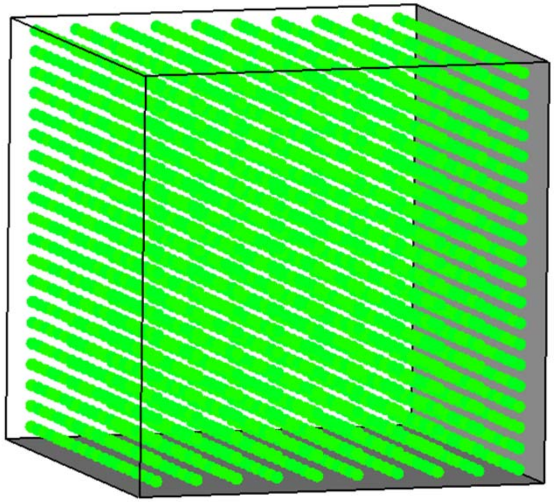


Is replaced by one here, shifted over toward the edge of the cell

Shift distance = γLt

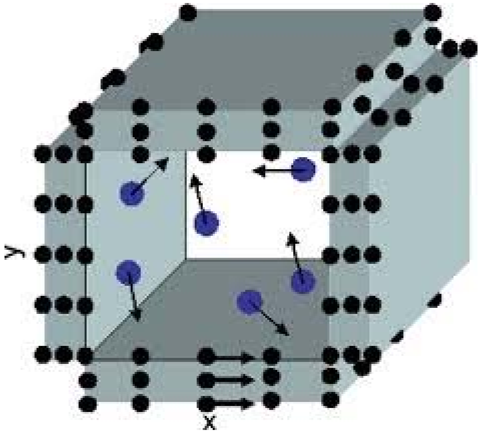
$$\gamma = \frac{dv_x}{dy}$$

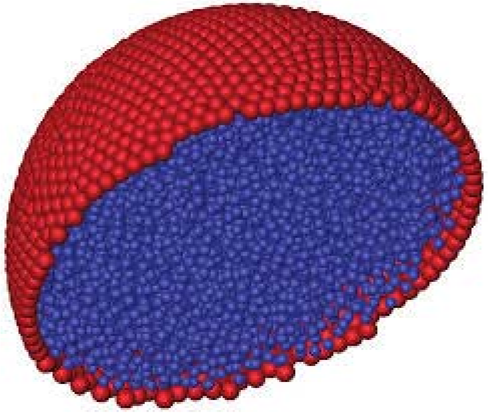


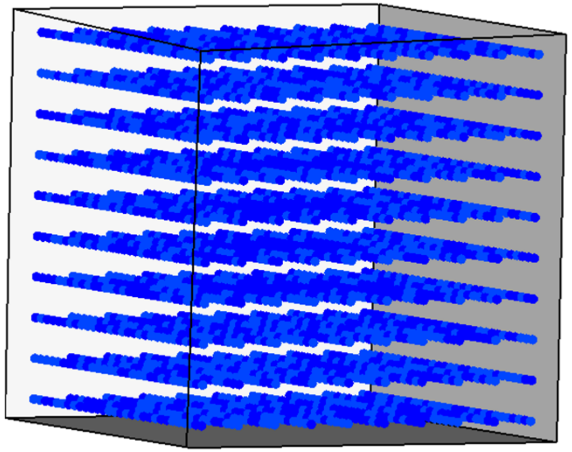


Implementation of Boundary conditions: wall

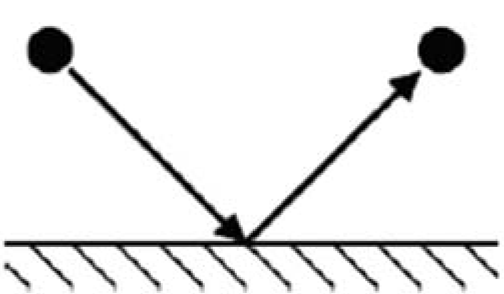
Solid objects in DPD are made of frozen particles



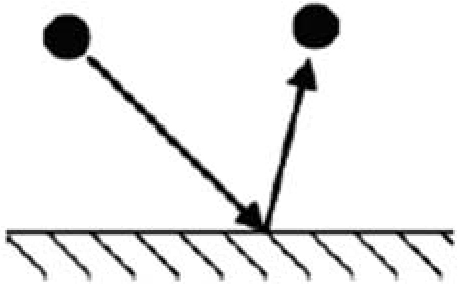




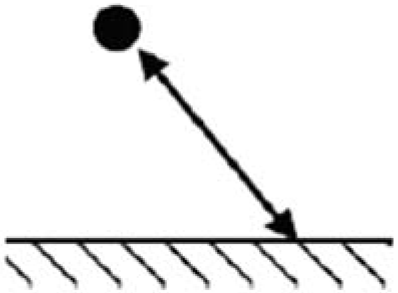
Soft repulsion in DPD cannot prevent particles from penetrating walls



Specular Reflection



Maxwellian Reflection

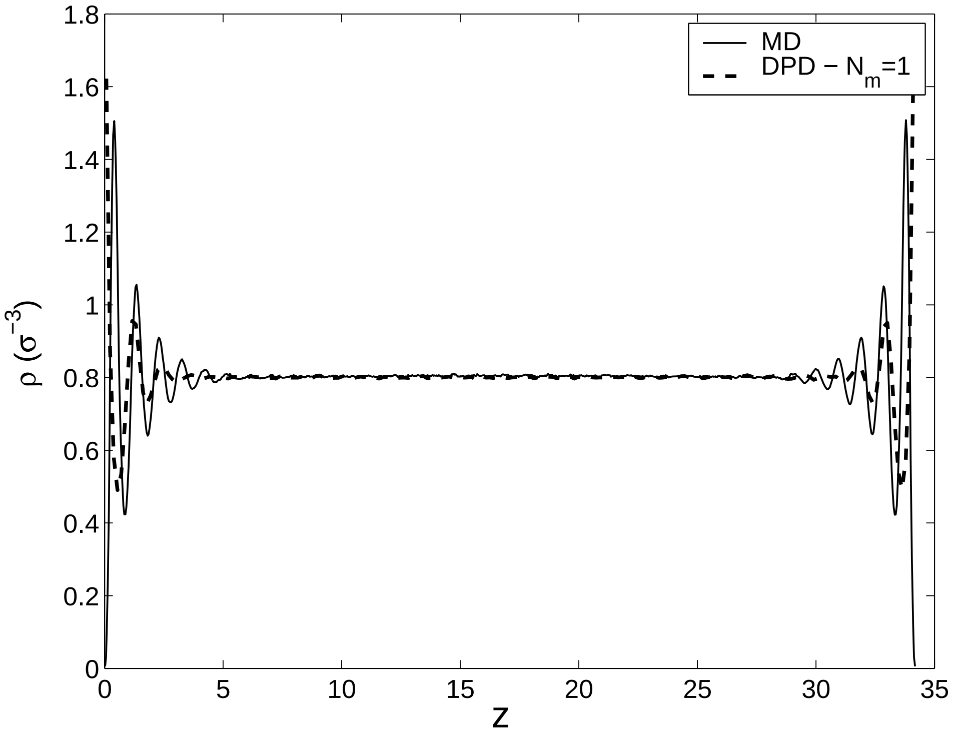


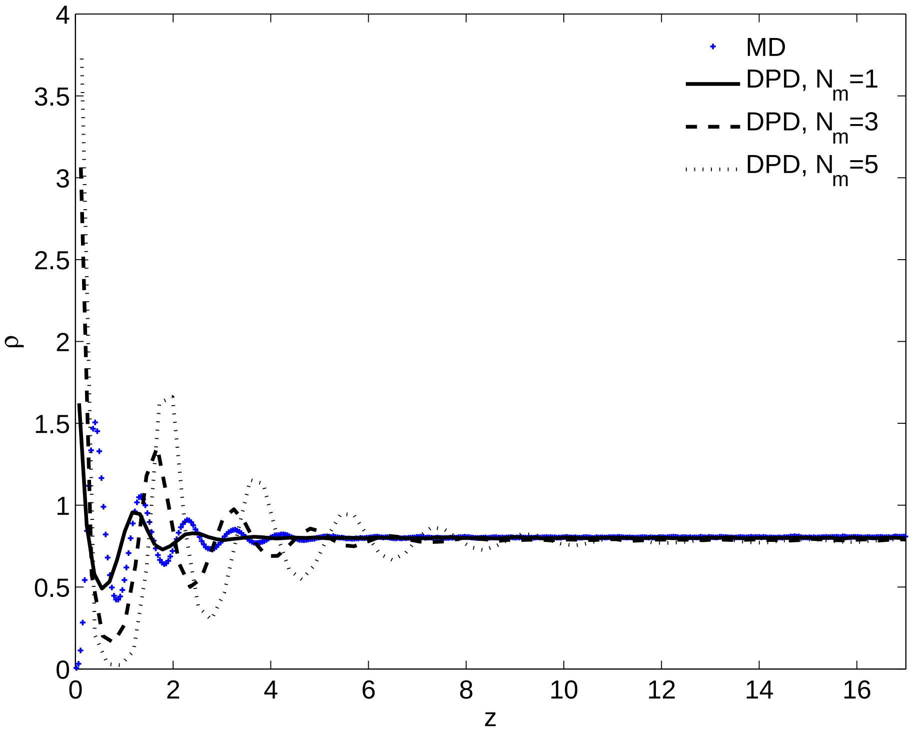
Bounce-back Reflection

Revenga, M., I. Zuniga, and P. Espanol, *Boundary Model in DPD.* International Journal of Modern Physics C, 1998.

Implementation of Boundary conditions: wall

Density Fluctuations in MD and DPD





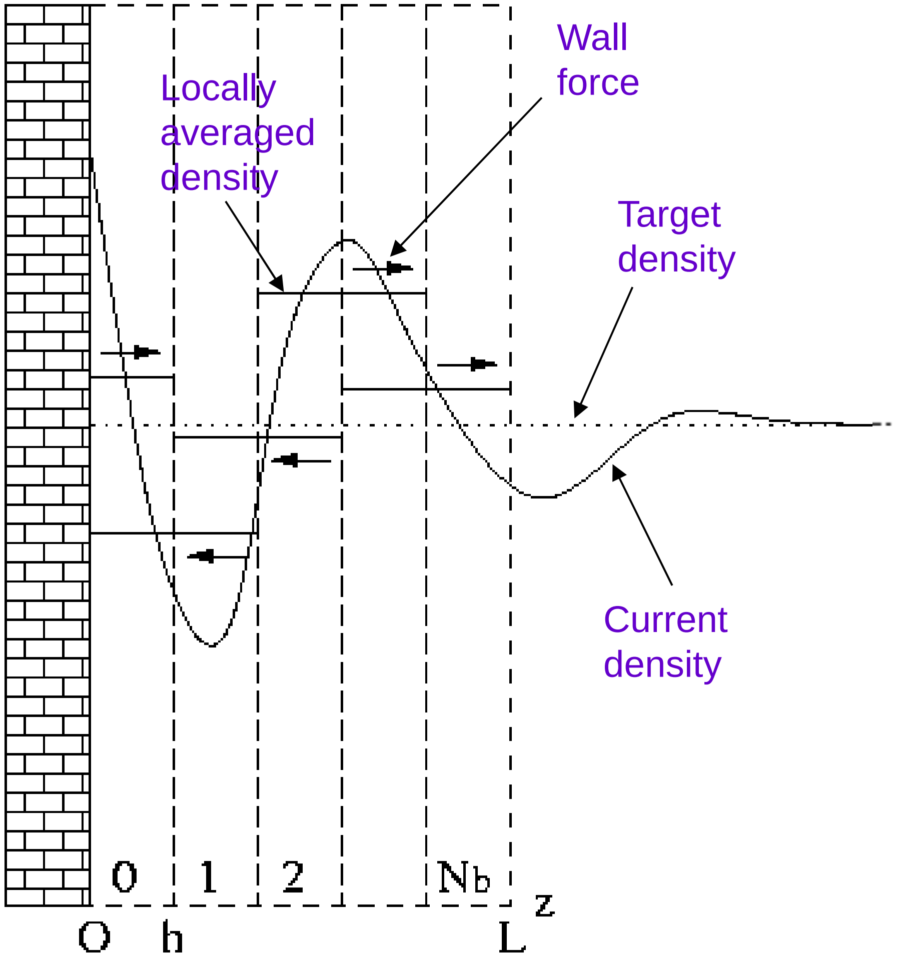
As coarse-graining parameter N_m increases, the density fluctuations in the DPD simulations also increase, with the N_m=5 case exhibiting very large values at the wall and also inside the flow domain.

This is not a desired effect, because we expect the fluctuations to decrease as we approach the continuum, i.e. $N_m \rightarrow \infty$.

E.E. Keaveny, I.V. Pivkin, M. Maxey and G.E. Karniadakis, J. Chem. Phys., 123:104107, 2005

Implementation of Boundary conditions: wall

Adaptive Boundary Conditions

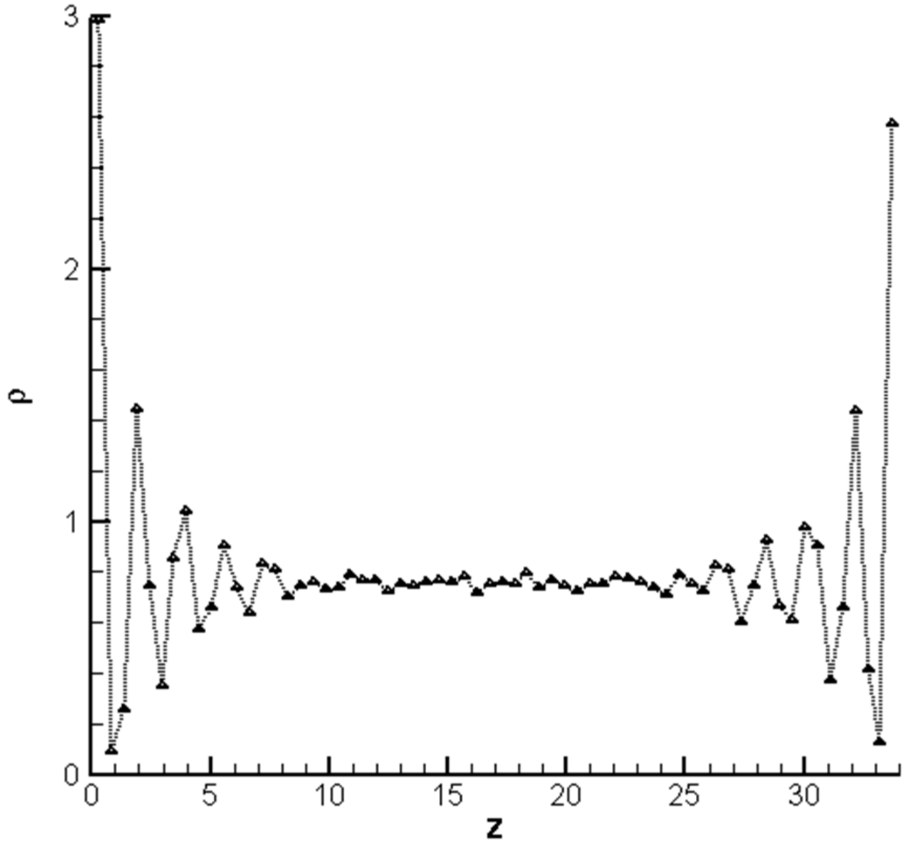


Locally averaged density

Wall force

Target density

Current density



Iteratively adjust the wall repulsion force in each bin based on the averaged density values.

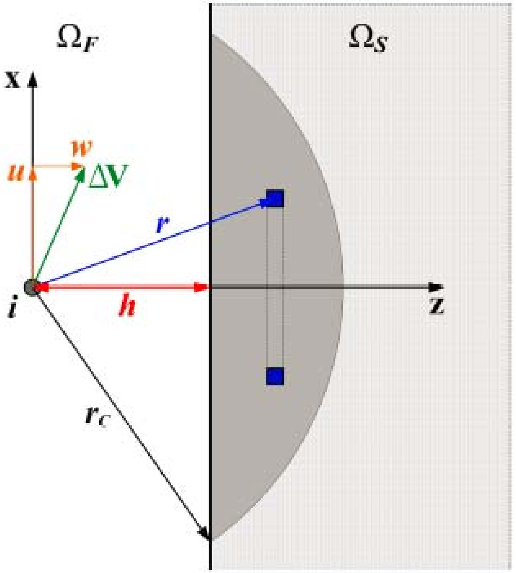
Adaptive BC:

- layers of particles
- bounce back reflection
- adaptive wall force

I.V. Pivkin and G.E. Karniadakis, PRL, vol. 101, 2008.

Implementation of Boundary conditions: wall

Effective boundary forces



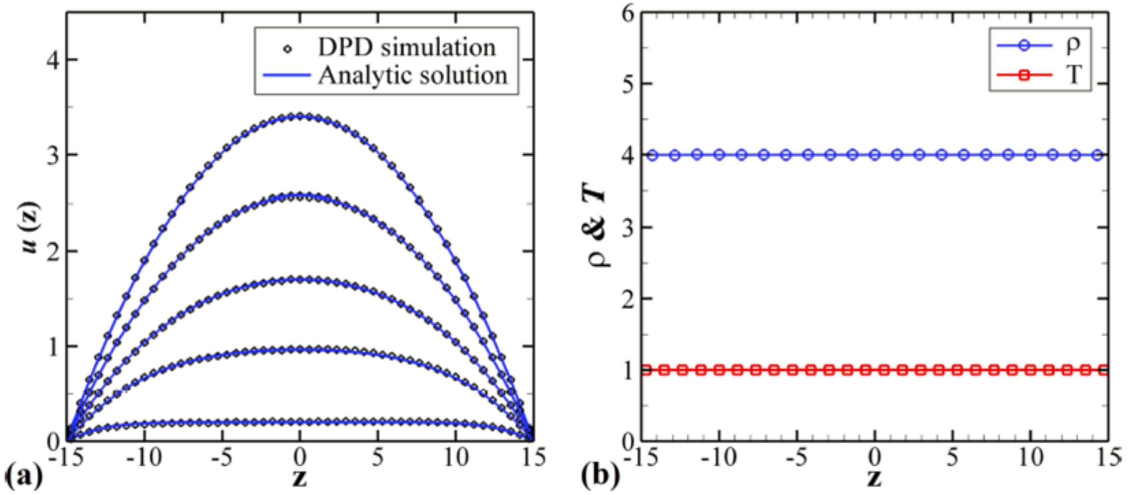
$$\mathbf{F}_C(h) = 2\pi\rho\mathbf{n}_w \int_{z=h}^{r_C}\int_{x=0}^{\sqrt{r_C{}^2-z^2}} F^C(r)g(r)\frac{z}{r}x\cdot dx\cdot dz = f^C(h)\mathbf{n}_w$$

$$\mathbf{F}_{D\|}(h) = -\pi\rho\gamma u\mathbf{e}_x \int_{z=h}^{r_C}\int_{x=0}^{\sqrt{r_C{}^2-z^2}} \omega_D(r)g(r)x^3\frac{z}{hr^2}\cdot dx\cdot dz$$

$$\mathbf{F}_{D\perp}(h) = -2\pi\rho\gamma w\mathbf{e}_z \int_{z=h}^{r_C}\int_{x=0}^{\sqrt{r_C{}^2-z^2}} \omega_D(r)g(r)xz^2\frac{z}{hr^2}\cdot dx\cdot dz$$

$$\mathbf{F}_R(h) = (\sigma_{R\|}(h)\mathbf{e}_x + \sigma_{R\perp}(h)\mathbf{e}_z)\,\xi$$

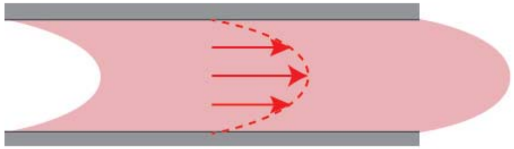
Time-evolution of the velocity profile in Poiseuille flow



Computing viscosity, diffusivity

Viscosity

Plane Poiseuille flow

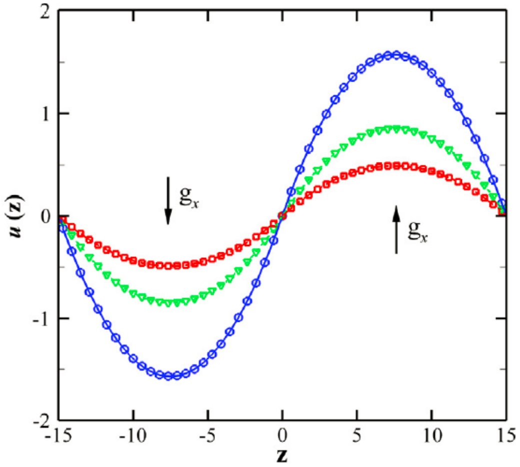


$$\frac{du}{\mathrm{dt}} = -\frac{1}{\rho}\frac{\mathrm{dp}}{\mathrm{dx}} + g_x + \nu\frac{d^2u}{dz^2}$$

Steady case and driven by g_x:

$$-\nu\frac{d^2u}{dz^2} = g_x$$

Periodic Poiseuille flow



$$-\nu\frac{d^2u}{dz^2} = g_x$$

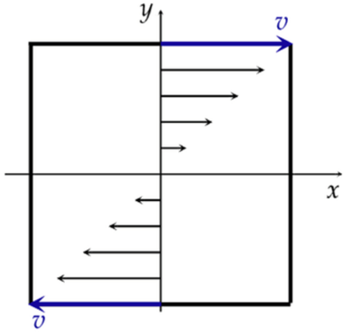
BC: $u(z=0) = 0$

$u(z=\pm d) = 0$

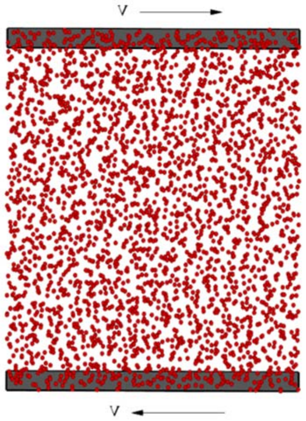
Solution:

$$u(z) = \frac{g_x}{2\nu} z(d - |z|)$$

Couette flow (constant shear rate)



$$\tau_{yx} = \mu\frac{du}{dy} = \mu\dot{\gamma}$$



$$J = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{bmatrix}$$

$$J_{\alpha\beta} = m\sum_{i}^{N} v_{i\alpha}v_{i\beta} + \frac{1}{2}\sum_{i\neq j}^{N} r_{ij\beta}F_{ij\alpha}$$

Diffusivity

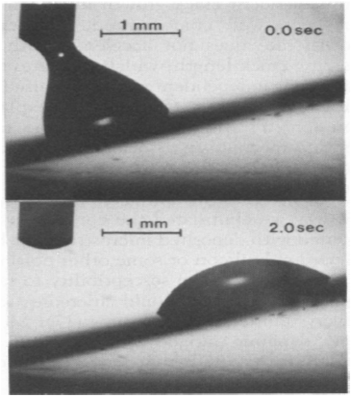
$$D = \lim_{t\to\infty}\frac{1}{6t}\left\langle \left|\mathbf{r}(t) - \mathbf{r}(0)\right|^2 \right\rangle$$

$$D = \frac{1}{3}\int_0^\infty \langle \boldsymbol{v}(t)\boldsymbol{v}(0)\rangle dt$$

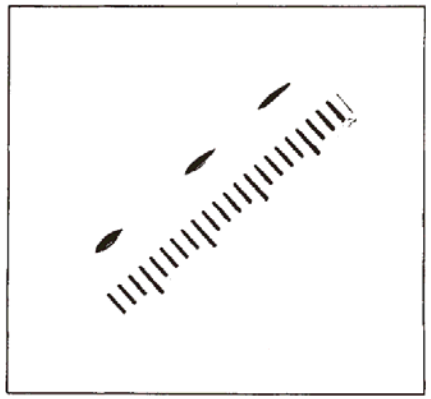
Part II: Dissipative Particle Dynamics

- **Brief introduction of DPD**
  - motivation, history
- **Foundations of DPD**
  - Coarse-graining of MD (Mori-Zwanzig formalism)
- **Parameterization of a DPD system**
  - Force field of classic DPD
  - Pressure, compressibility, viscosity, diffusivity of DPD fluids
  - Scaling
  - Modeling of complex fluids using DPD
- **How do you practically run a DPD simulation**
  - Implementation of boundary conditions
  - Computing viscosity, diffusivity
- **Some Applications**
  - Droplet, Blood flows, Self-assembly Dynamics

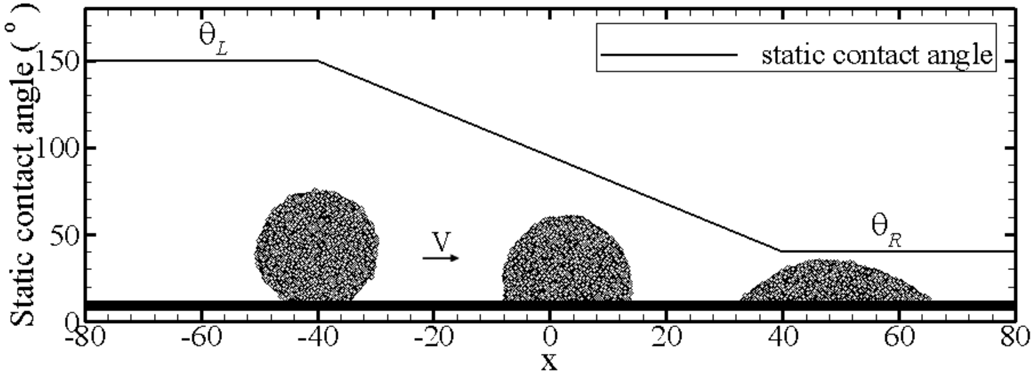
Some Applications: Droplet

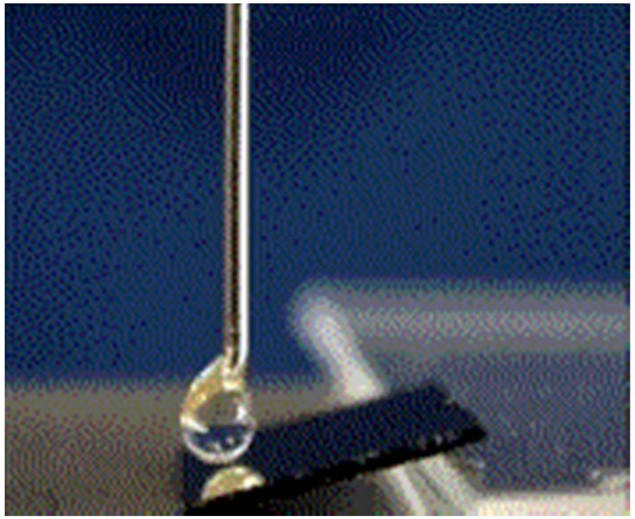


Chaudhury, et al. Science, 1992.

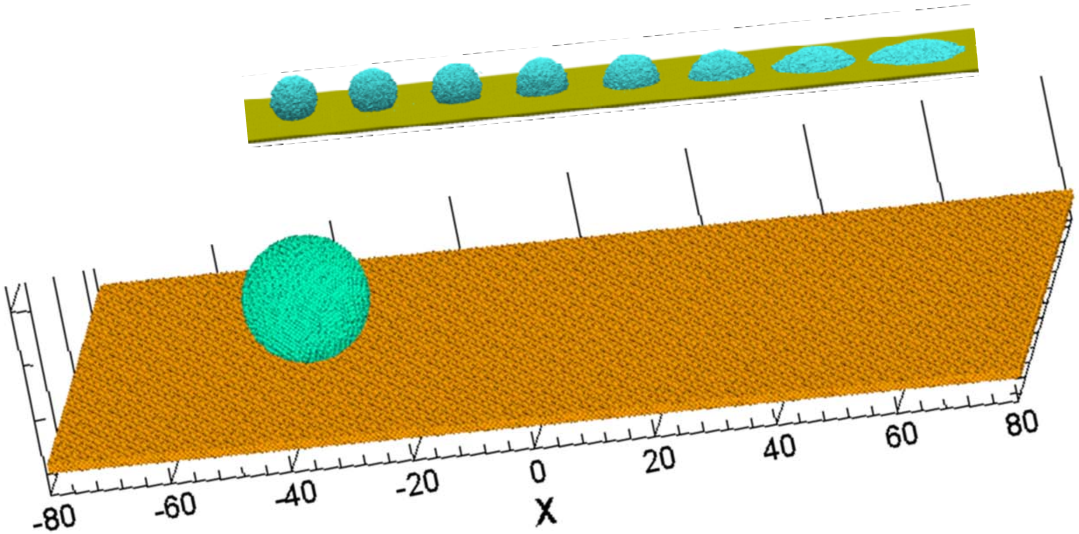


Bain, et al. Nature, 1994.



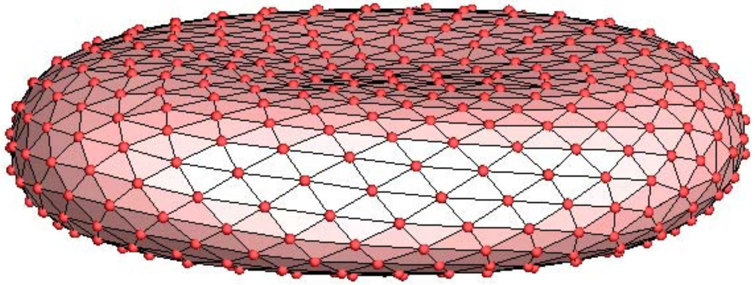


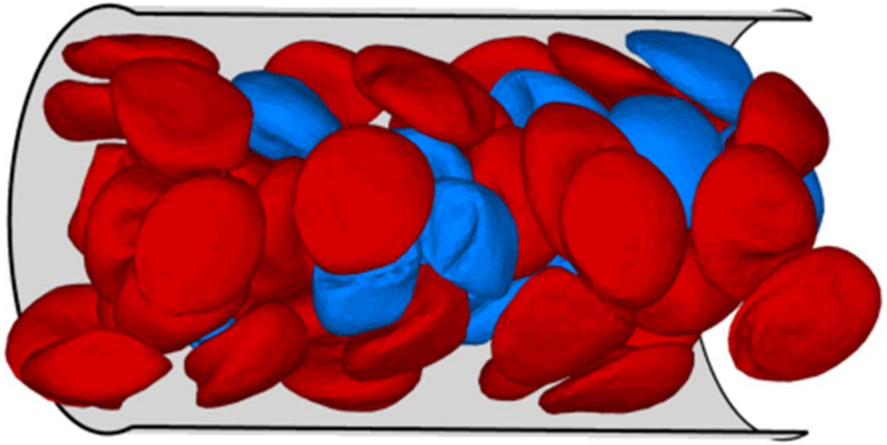
From the website of LSST of ETH Zürich

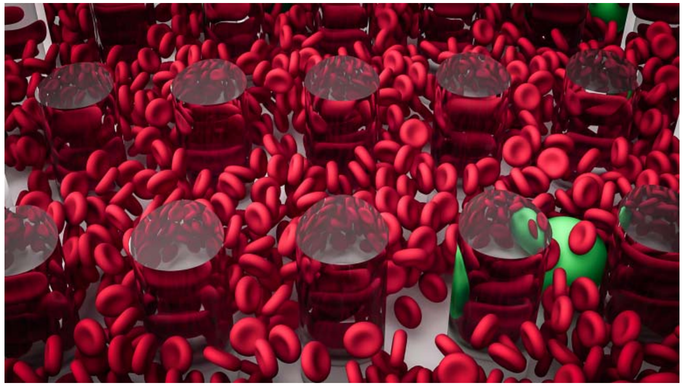


Li Z. et al., Phys. Fluids, 2013.

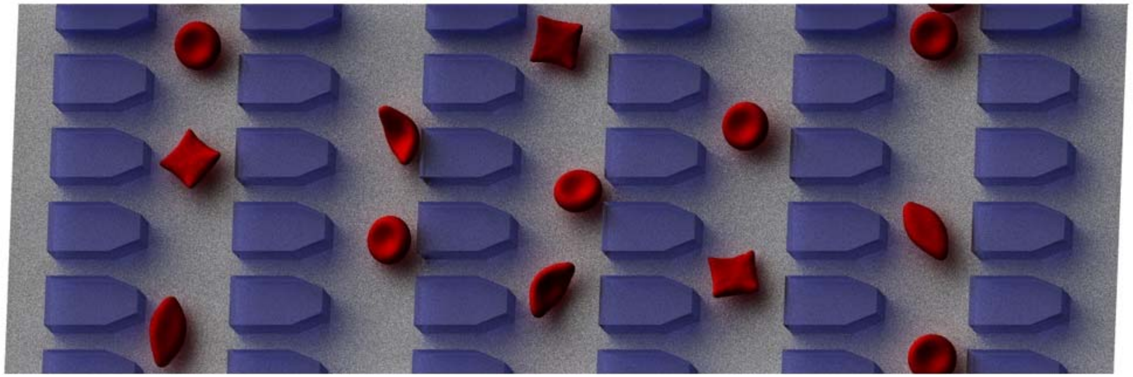
Some Applications: Blood flows





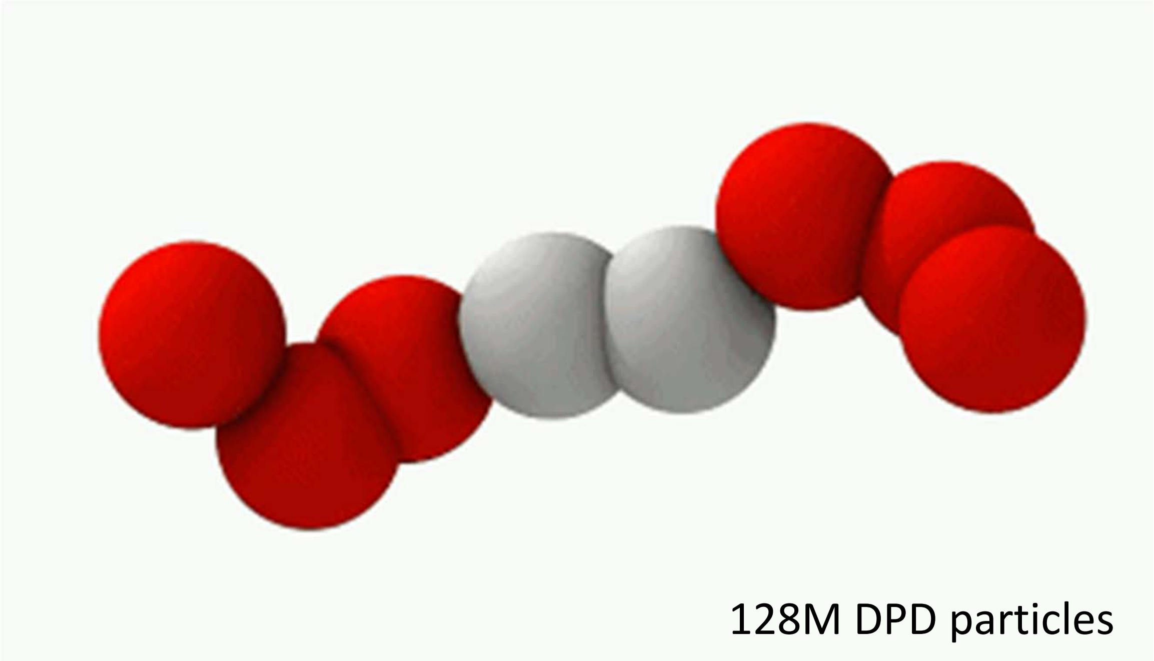


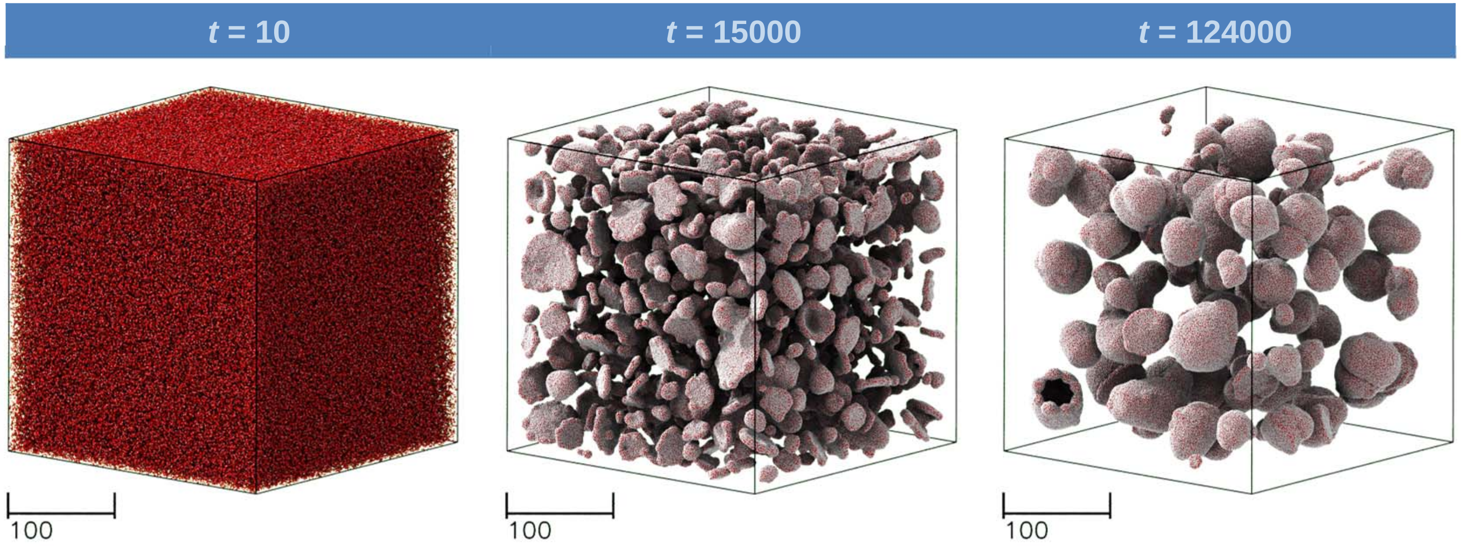
http://www.dam.brown.edu/people/ytang/index.php



Provided by Xuejin Li and Lu Lu of CRUNCH Group

Some Applications: Self-assembly Dynamics





GPU simulations performed by Yu-Hang Tang of CRUNCH Group

Software/Package for DPD simulation

LAMMPS: Large-scale Atomic/Molecular Massively Parallel Simulator

http://lammps.sandia.gov/

ESPResSo: Extensible Simulation Package for Research on Soft matter http://espressomd.org/

HOOMD-blue: a general-purpose particle simulation toolkit

http://codeblue.umich.edu/hoomd-blue/

DPDmacs: Fast coarse-grained simulations

http://www.apmaths.uwo.ca/~mkarttu/dpdmacs.shtml

Features: compatible with Gromacs.

MyDPD: C++ mesodynamics code

http://multiscalelab.org/mydpd

Features: simple, serial but functional.