LAMMPS Tutorial - Brownian Dynamics Simulation

Introduction to BD and LAMMPS

February 4, 2025

UNIVERSITY OF TURKU

Outline

1. Understanding Brownian Dynamics simulation

2. Studying self-assembly using the Langevin and Brownian dynamics

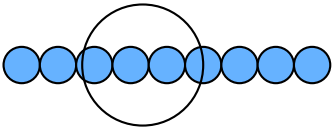
3. Performing simple thermodynamic calculations

Understanding Brownian Dynamics simulation

Understanding BD: A Look Back

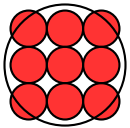
Advantage of not breaking the Symmetry at Mesoscopic Scale:

A sheet of water molecule



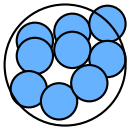
Isotropic thin film

Density dependent potential applied on Hard Core (BD)



Anisotropic minimum energy state with broken symmetry

Density dependent potential applied on Soft core (DPD)



An isotropic fluid droplet

Understanding BD: A comparison between the Hard Core and Soft Core models

Hard Core Potential

1. Symmetry breaking requires nucleation and growth.
2. System can stuck in a meta-stable phase, due to kinetic stability.

Soft Core Potential

1. System generally does not get stuck in the metastable phase.
2. Metastability have purely thermodynamic nature.
3. Not suitable for the nucleation and growth kind of phenomenon, where kinetically driven metastability plays an important role.

Understanding BD: A Look Back

Application of Soft Core model:

It can be used to model a system, where surface energy plays an important role, e.g, surface tension.

It can be applied for the mass transfer through convection.

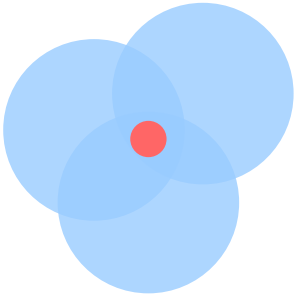
It can be applied for the heat transfer, where the particles have some internal degree of freedom.

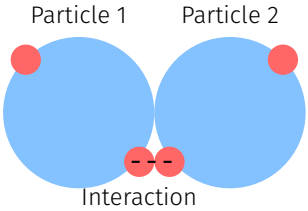
Understanding BD: A Look Back

Disadvantage of not breaking the Symmetry:

Let us consider a directional force defined by the potential

$$\chi_{ij} = \begin{cases} 1 & |\hat{\mathbf{r}}_{ij}.\hat{\mathbf{n}}_i| \geq cos(\Omega^\circ), \\ 0 & Otherwise \end{cases}$$





Patchy particle model
Treating directional forces

Soft-core potential
It does not serve the purpose

Hard-core potential
Serves the purpose

Understanding BD: Recap

- A large molecule is considered as a single sphere with a Hard Core and some interaction site.
- Preferred for the directional and selective bonding, e.g, self assembly in biologically inspired materials.
- Highly used in the NVT ensemble modeling.
- It does not care about the hydrodynamics.
- However, it can be coupled with other hydrodynamic based methods like MPCD to simulate the micro-swimmers like sperm cells.

Understanding BD: Recap to teaching session

Langevin equation for discrete models:

- **Langevin Equation:** Includes inertial effects to provide a microscopic discrete model for stochastic dynamics:

$$m\frac{d\mathbf{v}(\mathbf{r},t)}{dt} = -\gamma\mathbf{v}(\mathbf{r},t) + \boldsymbol{\xi}(t) - \nabla\Phi_i(\mathbf{r}).$$

Both the discrete (Langevin) and probabilistic whole-picture (Fokker-Planck) models stem from the foundational principles of Wiener processes. The Zwanzig-Mori formalism provides the theoretical core for deriving these equations.

Understanding BD: Recap to teaching session

Relation with the continuum model like Fokker-Planck equation:

- **Fokker-Planck Equation:** Describes the time evolution of the probability density function $P(x,t)$ in a continuous probabilistic framework:

$$\frac{\partial P(x,t)}{\partial t} = -\frac{\partial}{\partial x}\left[\mu(x)P(x,t)\right] + \frac{\partial^2}{\partial x^2}\left[D(x)P(x,t)\right].$$

Both the discrete (Langevin) and probabilistic whole-picture (Fokker-Planck) models stem from the foundational principles of Wiener processes. The Zwanzig-Mori formalism provides the theoretical core for deriving these equations.

Let us Do and check the equivalence of both the formalism through Langevin Dynamics.

Observing the evolution of density distribution in continuum model:

1. Rename your old git cloned tutorial folder so that you can download the fresh one.
2. Download all the codes from the website:
   `https://github.com/vikkivarma16/lammps_tutorial`.
   Command line:

```
git clone https://github.com/vikkivarma16/lammps_tutorial
```

Let us Do and check the equivalence of both the formalism through Langevin Dynamics.

Observing the evolution of density distribution in continuum model:

1. Go to the folder **`t2_bd/t2_langevin_focker_planck`**.
2. Run the Python code named **`t2_evolution_focker_planck.py`**.
3. Open the created Gif file and observe the evolution of the probability distribution of the particles going through Brownian dynamics, which is based on the Fokker-Planck formalism.

Let us Do and check the equivalence of both the formalism through Langevin Dynamics.

Observing the evolution of density distribution in discrete model:

1. Run you LAMMPS file named **`t2_langevin.in`**.
2. Load your output file **`simulation_data.lammpstrj`** into Ovito.
3. Select the **`add modification`** and **`histogram`**, and chose **`position_X`**.
4. Tick and set the range between $0 - 20$ for the **`x-axis`**.
5. Observe the evolution of the histogram and compare, it with the Fokker Planck probability distribution evolution.

Studying self-assembly using the Langevin and Brownian dynamics

Creating Patchy Particle Systems Using LAMMPS: Step 1

Step 1: LAMMPS Setup

- First, specify the units and atom style for the simulation.
- Use Lennard-Jones (LJ) units for the simulation. This is common for molecular dynamics simulations.
- Define the atom style as `hybrid sphere molecular dipole`, allowing different types of particles and interaction potentials.

LAMMPS Code

```
units lj
atom_style hybrid sphere molecular dipole
boundary p p p
```

Creating Patchy Particle Systems Using LAMMPS: Step 2

Step 2: Define Pair Styles

- The `pair_style` command defines the interaction potential between the particles.
- Use a hybrid pair style, combining both a Lennard-Jones potential and a cosine-squared potential for patchy interactions.
- The first potential describes the core-core interaction, while the second governs the interaction for peripheral (patchy) atoms.

LAMMPS Code

```
pair_style hybrid lj/cut 2.0 cosine/squared 0.12
```

Creating Patchy Particle Systems Using LAMMPS: Step 3

Step 3: Set Up the Simulation Box

- Use the `region` command to define the simulation box.
- The box dimensions are set to a cubic region of size 30.0 in each direction.
- Use the `create_box` command to initialize the box and define two atom types (core and patchy).

LAMMPS Code

```
region box block 0 30.0 0 30.0 0 30.0
create_box 2 box
```

Creating Patchy Particle Systems Using LAMMPS: Step 4

Step 4: Define Pair Coefficients

- Set the core-core interaction using the Lennard-Jones potential with specific parameters for `d_ii` and `epsilon`.
- The core-patch interaction is set to “none” to avoid direct interaction between core particles and patchy particles.
- The patchy-patchy interaction is modeled using a cosine-squared potential with the specified parameters.

LAMMPS Code

```
pair_coeff 1 1 lj/cut 0.01 1.3 2.0
pair_coeff 1 2 none
pair_coeff 2 2 cosine/squared 8 0.3 0.35
```

Creating Patchy Particle Systems Using LAMMPS: Step 5

Step 5: Define Molecule and Create Atoms

- Use the `molecule` command to define a molecule template (`patchy_part`) from the file `patchy_particle.mol`.
- Use `create_atoms` to randomly distribute 1000 patchy particles within the simulation box.

LAMMPS Code

```
molecule patchy_part patchy_particle.mol

create_atoms 0 random 1000 87910 NULL mol patchy_part 45475 overlap 1.5 maxtry 50
```

Creating Patchy Particle Systems Using LAMMPS: Step 6

Step 6: Set Atom Properties

- Set the mass and diameter for the two types of atoms.
- Type 1 (core) atoms are given a mass of 1.0 and a diameter of 1.0.
- Type 2 (patchy) atoms are given a very small mass (10^{-6}) to ensure they don't affect the dynamics too strongly and have no diameter set (effectively a "point" particle).

LAMMPS Code

```
set type 1 mass 1.0
set type 2 mass 0.000001
set type 1 diameter 1.0
set type 2 diameter 0.0
```

Creating Patchy Particle Systems Using LAMMPS: Step 7

Step 7: Group Particles

- Use the `group` command to categorize particles into groups: `core` for type 1 atoms, and `patch` for type 2 atoms.
- The `rigid_molecule` group is defined for particles of both types, ensuring they move rigidly together.

LAMMPS Code

```
group core type 1
group patch type 2
group rigid_molecule type 1 2
```

Creating Patchy Particle Systems Using LAMMPS: Step 8

Step 8: Define Fixes and Compute Thermodynamic Quantities

- Apply a Langevin thermostat to the core particles using the `fix thermo_stat` command. This controls the temperature of the system.

LAMMPS Code

```
fix thermo_stat core langevin 1.0 1.0 0.1 428984 omega yes compute kinetic_core core ke
fix kinetic_output core ave/time 100 1 100 c_kinetic_core file kinetic.dat mode scalar

compute temp_core core temp/sphere
fix temp_output core ave/time 100 1 100 c_temp_core file temperature.dat mode scalar
```

Creating Patchy Particle Systems Using LAMMPS: Step 9

Step 9: Set Simulation Parameters and Run

- Define the timestep (e.g., 0.005) for the simulation.
- Set the `thermo` output interval and specify the thermodynamic quantities to be printed.
- Finally, run the simulation for a specified number of timesteps (e.g., 100,000).

LAMMPS Code

```
timestep 0.005
thermo 100
thermo_style custom step temp ke pe press c_kinetic_core c_temp_core
run 100000
```

Performing simple thermodynamic calculations

Calculating the liquid-gas coexistence density:

Let us simulate LJ fluid and calculate the liquid-gas coexistence density following the instruction given in `Model 3` folder named **`T2E2_Instructions`**.

Conclusion

Conclusion:

- You have visualize the relation between discrete model and continumm models.
- You have successfully set up a patchy particle system using LAMMPS to study the self assembly.
- You have calculated a thermodynamic quantity using Brownian dynamics simulation

Thank you!

Questions and discussion welcome.

Learn by Example

Congratulations you have completed all the tasks !

——— May the force be with you ! ———