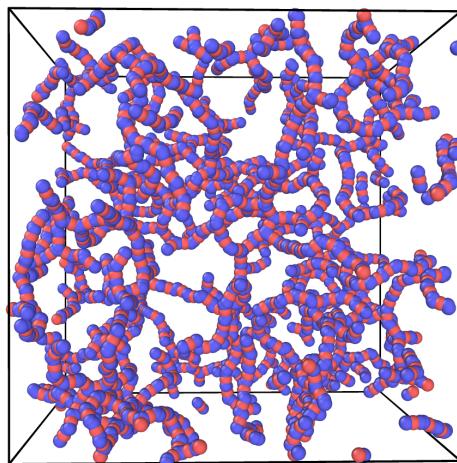


MTEK0033 Multiscale Modelling, University of Turku,
January 2026

Brownian Dynamics Simulation
Project- I



Created by: Vikki Anand Varma
Supervised by: Prof. Alberto Scacchi

If you come across any problems, please email at vavarm@utu.fi

Table of Contents

1	Miscellaneous strategies to compute transport properties in soft-matter system	2
1.1	Introduction	2
1.2	Brownian Dynamics Model	2
1.3	Systems Studied	3
1.3.1	(A) Lennard-Jones Fluid	3
1.3.2	(B) Polymer Beads with Bending Stiffness	3
1.3.3	(C) Hard-Sphere-Like Beads with Harmonic Bonds	3
1.4	Symbols	4
1.5	Methods for Viscosity Calculation	4
1.5.1	1. Green-Kubo Method	4
1.5.2	2. Muller-Plathe Method	4
1.6	Project Tasks	5
1.7	Simulation Studies	5
1.8	Presentation and Project Report	10
1.9	Future Aspects	10
2	Project Report Guidelines	10
3	References	13

1 Miscellaneous strategies to compute transport properties in soft-matter system

This project is designed to introduce students to the numerical study of transport coefficients in soft matter systems using **Brownian Dynamics (BD)**, implemented through the **overdamped Langevin equation**. Students will use **LAMMPS** and **Python-based analysis** to compute viscosity for different model fluids using both **equilibrium** and **non-equilibrium** methods.

The focus of the project is to understand how microscopic interactions, molecular architecture, and flexibility influence macroscopic rheological properties such as viscosity. Three representative model systems will be studied:

- Simple Lennard-Jones (LJ) fluid
- Polymer melts with tunable bending stiffness
- Hard-sphere-like bead chains with harmonic bonds

1.1 Introduction

Transport coefficients such as viscosity, diffusivity, and thermal conductivity characterize how momentum, mass, and energy propagate through matter. In soft matter and biological systems, these properties emerge from thermal fluctuations and interactions at mesoscopic length scales, where inertial effects are often negligible.

In such regimes, **Brownian Dynamics** provides an efficient framework by integrating out solvent degrees of freedom and describing particle motion using overdamped Langevin equations. This approach is particularly suitable for colloids, polymers, and biomolecular assemblies immersed in viscous environments.

In this project, students will explore how viscosity can be computed using:

1. **Green-Kubo relations**, based on equilibrium stress fluctuations
2. **Muller-Plathe method**, based on imposed momentum exchange and steady-state velocity gradients

By comparing these two methods across different systems, students will develop a deep understanding of equilibrium versus non-equilibrium approaches to transport phenomena.

1.2 Brownian Dynamics Model

The overdamped Langevin equation governing the motion of particle i is given by

$$\gamma \frac{d\mathbf{r}_i}{dt} = \mathbf{F}_i + \boldsymbol{\xi}_i(t), \quad (1)$$

where γ is the friction coefficient, \mathbf{F}_i is the deterministic force arising from interactions, and $\xi_i(t)$ is a Gaussian random force satisfying

$$\langle \xi_{i\alpha}(t)\xi_{j\beta}(t') \rangle = 2k_B T \gamma \delta_{ij} \delta_{\alpha\beta} \delta(t - t'). \quad (2)$$

Since inertia is neglected, momentum is not conserved locally. Nevertheless, viscosity can still be measured using appropriately defined stress correlations or externally imposed momentum fluxes.

1.3 Systems Studied

1.3.1 (A) Lennard-Jones Fluid

Particles interact via the Lennard-Jones potential:

$$U_{\text{LJ}}(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right]. \quad (3)$$

This system serves as a reference model to benchmark viscosity calculations using BD.

1.3.2 (B) Polymer Beads with Bending Stiffness

Polymers are modeled as bead-spring chains with:

- Harmonic bonds

$$U_{\text{spring}}(x) = k(x_0 - x)^2, \quad (4)$$

allowing interpolation between flexible and semi-flexible polymers.

1.3.3 (C) Hard-Sphere-Like Beads with Harmonic Bonds

This model approximates hard spheres using repulsive LJ interactions. Chain flexibility is controlled via the harmonic bond constant, allowing students to explore how connectivity and stiffness affect viscosity.

1.4 Symbols

Symbol	Meaning
η	shear viscosity
γ	friction coefficient
T	temperature
N	number of particles
N_p	number of polymers
s_p	beads per polymer
k_θ	bending stiffness
σ_{xy}	shear stress
$\dot{\gamma}$	shear rate

Table 1: List of symbols used in the project.

Note: If you use your own symbol please verify

1.5 Methods for Viscosity Calculation

1.5.1 1. Green-Kubo Method

The Green-Kubo method provides an **equilibrium route** to transport coefficients by relating macroscopic response to spontaneous microscopic fluctuations. In this framework, the shear viscosity is obtained from the time autocorrelation function of the off-diagonal components of the stress tensor:

$$\eta = \frac{V}{k_B T} \int_0^\infty \langle \sigma_{xy}(t) \sigma_{xy}(0) \rangle dt. \quad (5)$$

Here, V is the system volume, T is the temperature, and σ_{xy} is the microscopic shear stress. The angular brackets denote an ensemble average, which in practice is approximated by a time average over a sufficiently long equilibrium trajectory.

In Brownian Dynamics, although momentum is not conserved locally, stress fluctuations still encode information about the viscous response. Students will compute the stress tensor from particle interactions and integrate the stress autocorrelation function numerically. Care must be taken to ensure adequate sampling, as the long-time tail of the correlation function can contribute significantly to the viscosity.

This method is particularly useful for validating viscosity in the linear-response regime and serves as a benchmark for non-equilibrium techniques.

1.5.2 2. Muller-Plathe Method

The Muller-Plathe method is a **non-equilibrium** technique for computing viscosity by imposing a controlled momentum flux and measuring the resulting velocity gradient. In this approach, the simulation box is divided into slabs along the gradient direction, typically the y -axis.

At regular intervals, the particle with the highest positive x -momentum in one slab is exchanged with the particle with the highest negative x -momentum in another slab. This artificial momentum exchange generates a steady-state shear flow without applying external forces.

The imposed momentum flux J_p is known exactly from the cumulative exchanged momentum per unit area and time. The resulting velocity profile $v_x(y)$ is measured by averaging particle velocities within each slab. The shear viscosity is then computed as

$$\eta = \frac{J_p}{\partial v_x / \partial y}, \quad (6)$$

where $\partial v_x / \partial y$ is the slope of the steady-state velocity profile.

This method is computationally efficient and robust, particularly for systems where equilibrium stress fluctuations are noisy. However, it requires careful verification that the imposed shear rate remains within the linear-response regime.

1.6 Project Tasks

- Implement overdamped Langevin dynamics in LAMMPS
- Validate equilibrium temperature and diffusion
- Compute viscosity using Green-Kubo relations
- Generate shear flow using Muller-Plathe method
- Compare equilibrium and non-equilibrium viscosity

1.7 Simulation Studies

Study Parameters

Study 1: Polymer Length vs. Viscosity using Green-Kubo

In this study, you will investigate how the polymer length affects the viscosity of the system while keeping the *total monomer concentration* approximately constant. The polymer length is denoted by s_p , which represents the number of beads (monomers) in a single polymer chain.

#	s_p
1	8
2	12
3	16
4	24

Table 2: Effect of polymer length on viscosity while keeping the total monomer concentration approximately constant.

To complete this study, please carefully follow the steps outlined below.

Step 1: Hard-chain polymers

1. Navigate to the folder:

Project_1/Viscosity_through_green_kubo/Hard_chains/

2. Inside this directory, you will find four subfolders named:

8, 12, 16, 24

Each folder corresponds to a different polymer length s_p .

3. Open any one of these folders and locate the file:

polymer_template.mol

4. Open this file and count the number of particles (beads) in a single polymer chain. This number corresponds to the polymer length s_p .

5. Next, open the LAMMPS input file:

hard_chains_input.in

6. In this input file, locate the following commands:

```
molecule pol polymer_template.mol
create_atoms 0 random N_p 87910 NULL mol pol 454756 overlap 1.5
maxtry 50
```

7. Here, N_p represents the number of polymer chains in the system. The total number of monomers is given by:

$$N = N_p \times s_p$$

8. The number density is defined as:

$$\rho = \frac{N}{V}$$

Adjust the value of N_p so that the polymer density remains approximately:

$$\rho \approx 0.006$$

9. After updating N_p , run the simulation.

10. Once the simulation is complete, compute and record the viscosity value.

11. Repeat steps 3–10 for all polymer lengths:

$$s_p = 8, 12, 16, 24$$

and note down the corresponding viscosity values.

Step 2: Soft-chain polymers

1. Repeat the same procedure described in Step 1 for soft-chain polymers.
2. The corresponding directory is:

`Project_1/Viscosity_through_green_kubo/Soft_chains/`

3. Ensure that the polymer density is again kept approximately constant at $\rho \approx 0.006$.

Step 3: Lennard-Jones (LJ) particles

1. Now repeat the viscosity calculation for Lennard-Jones particles.
2. Navigate to the folder:

`Project_1/Viscosity_through_green_kubo/Lj_particles/`

3. In this case, LJ particles are single-bead particles. Therefore:

- The polymer length is fixed at $s_p = 1$.
- You do *not* need to run simulations for multiple s_p values.

4. The number density is simply:

$$\rho = \frac{N}{V}$$

where N is the total number of LJ particles.

Step 4: Data collection

1. After completing all simulations, collect the viscosity values obtained for each system.

Step 5: Plotting and data storage

1. Store the viscosity values for the four polymer lengths in the file:

`hard_chains.txt`

for hard-chain polymers.

2. Store the viscosity values for soft-chain polymers in:

`soft_chains.txt`

3. Store the viscosity value obtained from the LJ particle simulation in:

`lj_particle.txt`

4. Finally, plot viscosity as a function of polymer length s_p and compare the results for hard chains, soft chains, and LJ particles.

Study 2: Viscosity via Momentum Transfer (Muller-Plathe Method)

In this study, you will compute the viscosity of the system using the *momentum transfer method*, also known as the Müller-Plathe method. Unlike the Green-Kubo approach used in Study 1, this method induces a controlled momentum flux in the system and measures the resulting velocity gradient. The viscosity is then obtained from the ratio of the imposed momentum flux to the generated velocity profile.

The system setup, polymer lengths, and density constraints remain the same as in Study 1. The only differences are the way viscosity is computed and how the results are extracted.

Step 1: Hard-chain polymers

1. Navigate to the directory:

```
Project_1/Viscosity_through_momentum_transfer/Hard_chains/
```

2. As before, you will find four subfolders named:

8, 12, 16, 24

corresponding to different polymer lengths s_p .

3. In each folder, open the file:

```
polymer_template.mol
```

and verify the polymer length s_p by counting the number of beads.

4. Open the LAMMPS input file for hard chains and ensure that the number of polymer chains N_p is chosen such that the total monomer density remains approximately:

$$\rho = \frac{N_p \times s_p}{V} \approx 0.006$$

5. In this method, the viscosity is printed directly to the *LAMMPS log file*. Locate the following command in the input script:

```
thermo_style custom step temp f_4 v_dVx v_viscl f_vave
```

6. Here, `v_viscl` corresponds to the viscosity computed using the momentum transfer method.
7. Run the simulation. After completion, open the `log.lammps` file and record the reported viscosity value.
8. Repeat the above steps for all polymer lengths:

$$s_p = 8, 12, 16, 24$$

and note down the corresponding viscosity values.

Step 2: Soft-chain polymers

1. Repeat the same procedure described in Step 1 for soft-chain polymers.

2. The corresponding directory is:

`Project_1/Viscosity_through_momentum_transfer/Soft_chains/`

3. Ensure that the monomer density is again kept approximately constant at $\rho \approx 0.006$.
4. Extract the viscosity directly from the `log.lammps` file.

Step 3: Lennard-Jones (LJ) particles

1. Navigate to the folder:

`Project_1/Viscosity_through_momentum_transfer/Lj_particles/`

2. LJ particles are single-bead systems, so the polymer length is fixed at $s_p = 1$.
3. Only one simulation run is required.
4. The number density is given by:

$$\rho = \frac{N}{V}$$

where N is the number of LJ particles.

5. After the run, extract the viscosity from the `log.lammps` file.

Step 4: Velocity profile analysis

1. In each simulation folder, you will find a Python script named:

`plotter.py`

2. Use this script to plot the velocity profile generated by the momentum transfer method.
3. Verify that the velocity profile is linear in the shear direction, which confirms that the system has reached a steady-state shear flow.
4. Save the velocity profile plots for inclusion in your report.

Step 5: Data storage and plotting

1. Store the viscosity values for hard-chain polymers in:

`hard_chains.txt`

2. Store the viscosity values for soft-chain polymers in:

`soft_chains.txt`

3. Store the viscosity value obtained for LJ particles in:

`lj_particle.txt`

4. Plot viscosity as a function of polymer length s_p and compare the results obtained using the momentum transfer method with those from the Green–Kubo approach.

Study 3: Method Comparison

Compare the shear viscosity obtained using the Green–Kubo, Muller-Plathe methods for all model systems. Discuss agreements and deviations between the methods, with particular attention to finite-size effects, statistical uncertainty, and the validity of the linear-response regime.

1.8 Presentation and Project Report

- **Introduction (1 page):** Transport phenomenon in real life systems, it's examples and applications.
- **Model and Theory (1 page):** Viscosity Formula using Green-Kubo method, Basics of Mullar-Plathe and Shearing method to compute viscosity.
- **Results and Discussion (6–10 pages):** Viscosity calculation for different configuration and their comparison.
- **Conclusion (1 page):** Physical interpretation, limitations and future aspects.

1.9 Future Aspects

Discuss potential extensions of the present work, including the incorporation of hydrodynamic interactions, active particle dynamics, viscoelastic and non-Newtonian response, frequency-dependent viscosity, and coupling to explicit solvent models such as Dissipative Particle Dynamics (DPD) or Multi-Particle Collision Dynamics (MPCD).

2 Project Report Guidelines

1. Introduction to the Problem

This section should be approximately one page long (or slightly more) and should include five (minimum three) scientific references from peer-reviewed literature.

The introduction should clearly explain the concept of viscosity from both macroscopic and microscopic perspectives. In particular, students should:

- Introduce viscosity as a transport coefficient that quantifies momentum transfer in fluids.
- Discuss Newton's law of viscosity and the relationship between shear stress and velocity gradients.
- Explain how viscosity emerges from molecular-scale interactions and particle dynamics.
- Briefly describe how molecular dynamics simulations can be used to compute viscosity using:
 - equilibrium approaches (e.g., Green-Kubo relations),
 - non-equilibrium methods (e.g., shear flow or momentum transfer techniques).

The significance of viscosity should be discussed in relevant physical and engineering contexts, such as:

- transport in complex fluids and polymeric systems,
- nanoscale and microscale flows,
- biomedical and soft-matter applications where fluid rheology plays a critical role.

2. Technical Description of the Simulations

This section should describe the simulation methodology used in the project, focusing on how viscosity was computed in each study.

Students must:

- Include all plots generated in the project, corresponding to the different viscosity studies performed.
- Clearly state which simulation parameters were varied in each study (e.g., polymer length, density, interaction potential, strain rate, or thermostat parameters).
- Describe the ensemble used (NVE, NVT, or non-equilibrium setup) and justify its choice.
- Explain how velocity profiles, stress tensors, or time-correlation functions were obtained from the simulations.

While describing each plot, students should explicitly mention:

- the relevant control parameters,
- the physical quantity being measured,
- how the plotted quantity is related to viscosity.

3. Main Findings and Discussion

This section forms the core of the report and should be approximately 5–10 pages long (or 3–6 pages for a shorter report).

Students should:

- Present a detailed analysis of the results obtained in each study.
- Explain how viscosity depends on the system parameters explored in the project.
- Compare trends observed in different interaction models (e.g., hard chains, soft chains, Lennard-Jones particles).
- Discuss the consistency of the results with theoretical expectations or known physical behavior.
- Comment on sources of statistical uncertainty and numerical limitations.

Where appropriate, students should relate their observations to physical mechanisms such as:

- momentum transport,
- collisional versus correlated motion,
- effects of molecular structure and connectivity.

4. Conclusion

This section should be approximately half a page to one page long.

The conclusion should:

- Summarize the key findings of the viscosity studies.
- Highlight how molecular-level properties influence macroscopic viscous behavior.
- Reflect on the effectiveness of molecular dynamics simulations in predicting transport properties.
- Briefly suggest possible extensions of the project (e.g., temperature dependence, confinement effects, or non-Newtonian behavior).

References

- [1] Müller-Plathe, Florian. "**Reversing the perturbation in nonequilibrium molecular dynamics: An easy way to calculate the shear viscosity of fluids.**" Physical Review E 59.5 (1999): 4894.
- [2] **Computer Simulation of Liquids** by Allen and Tildesley.
- [3] **Theory of Simple liquids** by Hansen and McDonald.
- [4] **Nonequilibrium Statistical Mechanics** by Robert Zwanzig