Particle Methods
Spring Semester 2025

Homework 4

Due date: 30.04.2025

Write a program using language of your choice (C, C++, Matlab, python, ..) implementing Dissipative Particle Dynamics simulations in 2D with the following conditions:

- Assume the simulation domain is a 2D square, $L = 15$ units in size, periodic in all directions.
- Use DPD conservative, dissipative and random forces as specified in the Groot and Warren 1997 paper (copy on Teams), e.g.

$$\mathbf{F}_{ij}^C = \begin{cases} a_{ij}(1 - r_{ij}/r_c)\hat{r}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases},$$

$$\mathbf{F}_{ij}^D = -\gamma w^D(r_{ij})(\hat{r}_{ij} \cdot \mathbf{v}_{ij})\hat{r}_{ij},$$

$$\mathbf{F}_{ij}^R = \sigma w^R(r_{ij})\xi_{ij}\hat{r}_{ij},$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, $r_{ij} = |\mathbf{r}_{ij}|$, $\hat{r}_{ij} = \mathbf{r}_{ij}/r_{ij}$ and $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$. ξ_{ij} are symmetric Gaussian random variables with zero mean and unit variance, independent for different pairs of particles and at different time steps; $\xi_{ij} = \xi_{ji}$ in order to satisfy momentum conservation. Conservative force coefficients, a_{ij}, for particles i and j of different types will be specified below.

The weight functions in dissipative and random forces, ω^D and ω^R, are given by

$$\omega^D(r_{ij}) = \left[\omega^R(r_{ij})\right]^2,$$

$$\omega^R(r_{ij}) = \begin{cases} 1 - r_{ij}/r_c & \text{for } r_{ij} \leq r_c, \\ 0 & \text{for } r_{ij} > r_c, \end{cases}$$

and

$$\sigma^2 = 2\gamma k_B T,$$

where $k_B T$ is the system temperature in non-dimensional units used in simulations.

Take $\sigma = 1$, $\gamma = 4.5$ and $r_c = 1$. Assume that all particles have the same mass $m = 1$.

- Use velocity-Verlet scheme to integrate equations of motion in time. Please pay attention to the $\sqrt{dt}$ term for the random force $\mathbf{F}^R$ in the DPD equations of motion!
- Use random distribution of particles in the domain as your initial conditions. Take the density of particles to be equal to $\rho = 4$, so the total number of particles in the domain should be equal to $\rho \times L^2$. Set the initial particle velocities to be equal to zero.
- Implement and use cell list to speed up your simulations. Take the size of the cell equal to $r_c = 1$.
- Use 4 different types of particles in simulations, i.e. F (fluid), W (wall), and two additional types of particles A and B for modeling of molecules.
- To simulate molecules, implement calculations of bonds (springs). We will use only one type of bond in all simulations. For particles i and j connected by the bond, the harmonic spring force should be equal to

$$\mathbf{F}_{ij}^S = K_S(1 - r_{ij}/r_S)\hat{r}_{ij},$$

where r_S is the equilibrium bond length and K_S is the bond constant.

- Implement modeling of 2 types of molecules, a chain molecule and a ring molecule. Each chain molecule shuold consist of 7 particles connected with bonds, 2 particles of A type and 5 particles of B type, i.e. $A - A - B - B - B - B - B$. Each ring molecule should consist of 9 particles of A type connected by bonds in a ring structure. All particles in the molecules should interact with other particles in the domain using usual DPD forces. Conservative force parameters and bond parameters will be specified below.

- To model solid walls, after initial random distribution of particles in the domain and before starting the simulations, choose all fluid particles (type F) that are located withing the wall region, and change their type to the wall particles (type W). During the simulations, the wall particles should interact with other particles using usual DPD forces, however their motion should be prescribed. If the wall is stationary, the positions of the wall particles should be fixed and their velocities set to 0. If the wall is moving at constant velocity v_{wall}, the wall particles velocity should be set to v_{wall} and their positions should be updated accordingly in simulations (motion with constant velocity). Set the width of the walls equal to cutoff radius $r_c = 1$.

- (optional) Implement bounce-back reflection at the surface of the wall. If you plan to do it, let me know, I can share with you some tricks to simplify implementation.

a) Perform preliminary test simulations containing only fluid particles without walls and with periodic boundary conditions. Set conservative force coefficient $a_{ij} = 25$ for fluid particles in the simulations. Use conservation of total momentum as one of the checks of correct implementation. Calculate temperature and make sure that it is close to the value defined by the DPD thermostat. Investigate dependence of temperature on the timestep used. For your reference, typical timestep used in DPD simulations of simple fluid is about 0.01.
b) Perform simulations of Couette flow with chain molecules. To model the flow place two walls on opposite sides of the domain, moving in opposite directions with constant velocity $v_{wall} = 5$. Initialize the system with 42 chain molecules located at random inside the domain. Add fluid particles at random, create walls. For the bonds, use $K_S = 100$ and $r_S = 0.1$. Set the conservative force coefficient a_{ij} for particles of different types equal to

$$a_{ij} = \begin{pmatrix} & A & B & F & W \\ A & 50 & 25 & 25 & 200 \\ B & 25 & 1 & 300 & 200 \\ F & 25 & 300 & 25 & 200 \\ W & 200 & 200 & 200 & 0 \end{pmatrix}.$$

Start your simulations and allow system to evolve. Describe the motion and distribution of molecules in the domain. Explain the results.
c) Perform simulations of Poiseuille flow with ring-molecules. To model the flow place two fixed walls on opposite sides of the domain, the velocity of the walls should be set to $v_{wall} = 0$. To drive the flow, at each time step, add to each non-wall particle in the domain a constant body force in the flow direction, $\mathbf{F}^{body} = 0.3$. Initialize the system with 10 ring molecules located at random inside the domain. Add fluid particles at random, create walls. For the bonds, use $K_S = 100$ and $r_S = 0.3$. Set the conservative force coefficient a_{ij} for particles of different types equal to

$$a_{ij} = \begin{pmatrix} & A & F & W \\ A & 50 & 25 & 200 \\ F & 25 & 25 & 200 \\ W & 200 & 200 & 0 \end{pmatrix}.$$

Start your simulations and allow system to evolve. Describe the motion of molecules in the domain. Run the simulations for a extended time, around 5000-10000 steps and look at the evolution of the distribution of the molecules in the domain (in cross-flow direction). (optional) What is the reason for the change of the distribution? (note that the answer to the last question may be not trivial, so you are welcome to do a quick search in the literature.)

Please submit your code together with the report.