Dissipative Particle Dynamics

In this assignment you will study the mesoscopic scale particle-based simulation technique called Dissipative Particle Dynamics (DPD). You are asked to change the Molecular Dynamics code into a DPD code for simulating polymeric liquids. As a reference the paper of Groot and Warren, J. Chem. Phys., 1997, v. 107, p. 4423 will be used. Please, use DPD units in your code: The unit of length is the cutoff distance of the DPD interactions, r_c. The unit of mass is the DPD-particle mass, m. Last, the unit of energy is kT.

Questions:

1. Implement the DPD method. Hints:

   - Start from the code handed out for the MD assignment, i.e. the source contained in `PBS-MD.zip`.
   - The structure of a DPD code is very similar to that of MD. You are advised to keep using the velocity-Verlet algorithm already implemented. This is equivalent to Eq. (9) in Groot and Warren with $\lambda = \frac{1}{2}$.
   - Take small steps when changing a code. After implementing a new feature perform small simulations and see if the code still produces sensible results. Visualization (by using Ovito) can provide lot's of insight.
   - The dissipative and random DPD interaction act together as a thermostat. Therefore you should not call any other thermostats such as the Berendsen thermostat.
   - The major changes in the code will be in `forces.c`. For all particle pairs within a cutoff distance of 1 there are three contributions to the DPD pair-force, namely, a conservative pair-force, a dissipative pair-force and a random pair-force. See Eqs. (2-4) of Groot and Warren. These pair-forces replace the Lennard-Jones interactions in the original MD code. Use the values $\gamma = 4.5$ and $\sigma = 3$, which are linked via Eq. (5).
   - Implement type-dependent non-bonded DPD interactions forces.
   - For the random force you need a random variable ζ_{ij} with mean zero and variance 1, see Eq. (8). In the first paragraph of the second column on page 4426, Groot and Warren discuss the usage of Gaussian vs. uniformly distributed numbers. You are recommended to use uniformly distributed numbers. Uniform numbers drawn from the interval $[-\sqrt{3}, \sqrt{3}]$ have mean 0 and variance 1. You can use the random number generator implemented in `random.c`.
   - Adapt your code such that you can define 'bead-spring' chains of a length N. The spring-force between two bonded beads obeys $\mathbf{f}_{ij} = -C\,\mathbf{r}_{ij}$, with $C = 2\,kT\,r_c^{-2}$ (i.e. 2 in DPD units). Note there seems to be a sign error in Eq. (25). Differently from atomistic MD, the spring force in DPD is always attractive. Therefore, to counteract excessive overlap between bonded pairs in DPD, 1-2 and 1-3 bonded pairs need

to be included in the non-bonded interactions. In the code this can be established by setting the `exclude_12_nb` and `exclude_13_nb` members of the `parameters` struct to zero.

- Also implement an initialization routine for a system consisting of bead-spring chains that avoids over-stretched springs).

2. Verify your implementation for $N = 1$:

   - Shut down the dissipative and random interactions and show that total energy is conserved. For this you also need to derive an expression for the potential energy that is, of course, related to the conservative pair-forces.
   - Shut down the conservative interactions, while keeping the combined dissipative and random interactions turned on. Show that without conservative interactions a DPD fluid does not have any spatial structure and is in that respect similar to an ideal gas.
   - Show that the combined dissipative and random interactions give the proper particle-velocity equilibrium statistics consistent with the imposed temperature, independent of the conservative interactions.

3. Validate your implementation for $N = 1$:

   - Reproduce the radial distribution function shown in Fig. 2.
   - For a binary mixture of DPD particles with the parameters provided in the caption of Fig. 6, produce a figure similar to Fig. 6.
   - Use Eq. (18) to 'measure' the χ-parameter from the simulation results. Perform a series of runs with varying a_{AB} and reproduce the dependency shown in Fig. 7 for $\rho = 3$.

4. Validate your implementation for $N > 1$:

   - Provide a snapshot of equilibrated bead-spring chains of length $N = 4$ (e.g. using Ovito).
   - Perform two-phase simulations for $N = 2$, 4 and 8. Provide plots for each of these 3 chain lengths, similar to Fig. 6 (with a suitable chosen value of a_{AB} for each case).
   - Find the χ parameters using Eq. (18) and compare the results with the correlation given by Eq. (27) and shown in Fig. (8).

Report

- Write a concise report that demonstrates you have implemented DPD correctly, by presenting the verification and validation results.
- Upload onto Canvas the report as pdf and source code(s) as zip containing all relevant `.h` and `.c` files.