User Manual of `DropDPD` Simulation Software

Eunsang Lee

March 31, 2021

1 Modules

`DropDPD` includes the main simulation module and analysis modules. The usage of the program is:

```
mpirun -np [number of cores] dropdpd [module name] [options],
```

or

```
dropdpd [module name] [options]
```

Here is an example of the main module of the software.

```
mpirun -np 8 dropdpd run -c conf.gro -p control.in -t topol.top -o impact
```

The names of modules are listed as follows:

`run`: Main dpd simulation module

`dropsize`*: Module for droplet diameter, base diameter, diameter of the second layer from the surface, contact angle, and height of the droplet as a function of time

`velocity`: Module for a droplet velocity in each direction as a function of time

`rdensity`: Module for an averaged radial density

`sphstress`*: Module for calculating spherical stress from the trajectory

`polads`: Module for the polymer adsorption on the surface as a function of time

`polsize`: Module for polymer size(end-to-end distance, radius of gyration...) as a function of time and its distribution averaged over all simulation time

`polevrlx`: Module for polymer end-to-end vector relaxation time

`bondlen`: Module for average bond length as a function of time

`polstretch`: Module for stretching amount of polymer as a function of time

`trjtogro`: Module for converting a binary trajectory to a gro file (ASCII)

`polsmsf`: Module for single molecule structure factor

`polsubsize`: Module for subchain sizes of polymer

`msd`: Module for mean square displacements

`surfcov`: Module for surface coverage

`rdf`: Module for radial distribution functions

`brdgsize`: Module for contact angles of a capillary bridge

`brdgvel`: Module for average velocities of particles in a capillary bridge

*: Only these modules support multi-core.

The main simulation module requires basically three input files, initial configuration (`.gro`), topology (`.top`), and control (`.in`)files. The analysis modules additionally require an output from the main module, trajectory (`.trj` or `.gro`)), force (`.frc`) or stress(`.str`) file depending on the module. One can use following options to specify input and output file names.

`-c`: (default: conf.gro) Input initial configuration file

`-t`: (topol.top) Input topology file

`-p`: (control.in) Input control file

`-o`: Prefix for output files

`-l`: (log.out) Output log file

`-x`: (traj.trj or traj.gro) Output trajectory file

`-f`: (force.frc) Output force file

`-s`: (stress.str) Output stress file

`-r`: (check.ckp) Checkpoint file to generate

`-restart`: Checkpoint file to restart simulation

`-ss`: (sslog.out) Output log file for slip-springs

2 Control variables

In this section, options for control variables are explained. Default values are given in the parenthesis. In the control file, the desired values have to be given next to the argument in a single line separated by a space (or a tab). Any values or strings after the character ";" is taken as a comment and is ignored.

- `temperature` (default: 1.0) The temperature of the simulation given to the DPD random force.
- `timestep` (0.02) The integration time step.
- `gamma` (4.5) The coupling parameter of the drag force in DPD simulation.
- `saferatio` (10.0) The multiplication factor to the number of beads in a domain for memory allocation.
- `maxforce` (100.0) The maximum force to stop energy minimization.
- `lambda` (0.5) The division factor for the estimation of the velocity at $t + \Delta t/2$ by DPD velocity-Verlet algorithm.
- `cellcutoff` (1.0) The cut-off distance for the cell-list. This has to be larger than the longest cutoff distance of the non-bonded interaction.
- `totalsteps` The number of integration steps for whole simulation.
- `dumpbinary` (yes) `yes`: Outputs are written binary if yes. `no`: Written in text format.
- `xtrjfreq` (100) A frequency to write a trajectory
- `xlogfreq` (1) A frequency to write a log
- `xstrfreq` A frequency to write stress of each particle
- `xfrcfreq` A frequency to write force of each particle
- `rmcomvreq` A frequency to remove center-of-mass motion
- `rmcomvdir` (x y z) A direction of the removal of the center-of-mass motion
- `randseed` A seed of a random number generator
- `dumpfrozen` (no) `no`: Data of frozen particles is not written in trajectory, stress, and force files. `yes`: It's written.
- `integrator`

vv: Velocity-Verlet algorithm

emin: Energy minimization

vv-sllod: SLLOD Velocity-Verlet algorithm (under deformation)

- `pullspringk` A constant of springs pulling all particles toward a fixed point in a simulation box. Followed by the direction of pulling in case it is not isotropic (x/y/z/xy/yz/xz)
- `pullcenter` (Center of the box) A position of the fixed point to which the springs are attached
- `gravity` External gravitational field. Followed by direction (x/y/z)
- `nonbonded` Potential function for non-bonded interactions

  dpd: Dissipative particle dynamics

  mdpd: Multi-body dissipative particle dynamics
- `bondlength` (harmonic) Potential function for bond length
- `wall` Wall boundary condition followed by direction (x/y/z)

  solid: Reflective boundary condition
- `wallposx/wallposy/wallposz` Position of two walls along z-direction
- `slipspring` (0) The total number of slip-springs
- `sscutoff` (1.5) The cutoff distance of slip-springs
- `seqnmcsteps` (0) The number of MC steps per each sequence
- `seqndpdsteps` (0) The number of DPD steps per each sequence
- `ssparam` (0.0 0.0) The spring constant and the average bond length of the slip-spring