Tutorial for Capillary Bridge Under Shear with a DPDWetting software

Eunsang Lee

June 20, 2022

This tutorial is for Multi-body dissipative particle dynamics simulations of a capillary bridge confined in two solid rough surfaces under steady shear. One can reproduce results in the journal article [Lee et al., J. Chem. Phys. DOI:10.1063/5.0098150] from the following tutorial.

1. Prereqisites

1.1. DPDWetting Software

You can download the DPDWetting software from git. To install, do the follwing commands. OpenMPI ver 3.1.1 or higher is required.

```
git clone https://github.com/elee-tud/dpdwetting.git
cd dpdwetting/build
make
```

1.2. Python tools

Python scripts enclosed in the directory, `dpdwetting/pylib` will be used to generate input files and post-process the output files.

2. MDPD simulation

A simulation of a capillary bridge under steady shear is done with the following four steps.

1. Generation of morphology of a capillary bridge confined in two rough surfaces
2. Energy minimization of the capillary bridge
3. Equilibration of the capillary bridge
4. Shear applied to the capillary bridge

2.1. Generation of an Initial Configuration

In this step, one generates a capillary bridge of a liquid with the total number of liquid beads 10^5. The separation between two walls is 30 excluding the pillar height, and the simulation box size along y is 12. The size along x is set big enough to be safe from finite size effect in shear simulations as $L_x = 120$. The liquid particles are initially placed in a cuboid between two surfaces with the size along x equals 46. This process can be simply done by the python script `generate_polroughfilm.py` enclosed. After moving to the directory `simulations`

```
generate_polroughfilm.py -s 100000 -x 120 -y 12 -z 30 -ix 46 -width 2 -gap 1
-height 2 -sw -25 -dir xy
```

This will generate a configuration file (`conf.gro` by default) and a topology file (`topol.top`) needed to run a simulation. Surface roughness and the strength of the solvent-surface interaction strengths can be varied by following options of this script

`-width`: Width of the pillars

`-gap`: Gap between the pillars

`-height`: Height of the pillars

`-sw` : Amplitute parameter of between solvent and surface interaction (default: -30)

`-dir`: Direction of pillar (x: x-striped surface, y: y-striped surface)

2.2. Energy minimization

Energy of the initial configuration is minimized by a steepest decent algorithm. A control file for the energy minimization is enclosed (`1_emin.in`). It contains information about the running parameters specified by `integrator emin`. One can run the minimization with the following command:

```
mpirun -np [number of cores] dpdwetting run -p 1_emin.in -o 1_emin
```

The simulation ends if the maximum of the particle force is less than 100 (by default, but can be modified). It will generate output files of the program, `1_emin.trj`, `1_emin.frc`, `1_emin.str`, `1_emin.out`, `1_emin.ckp`, and `1_emin_final.gro`. `1_emin_final.gro` is the final configuration which is used for the equilibration.

2.3. Equilibration

Equilibration is performed with another control file, `2_emin.in` which specifies the velocity-Verlet-like integrator by `integrator vv`. One needs to specify also the initial configuration file.

```
mpirun -np [number of cores] dpdwetting run -c 1_emin_final.gro -p 2_equil.in
-o 2_equil
```

From the trajectory produced in this simulation (`2_equil.trj`, one can calculate the particle density of the capillary bridge as a function of z-coordinate. To do so, use the following modules.

```
dpdwetting brdgzd -x 2_equil.trj -c 1_emin_final.gro -p 2_equil.in
```

Following modules can be used by the same way to calculate the equilibrium properties of the capillary bridge.

1. `dpdwetting brdgsize` : Contact angle of the capillary bridge (number of points to be fitted to a parabola, the height of each liquid slab, and the height of the pillars have to be given with '-np 4 -dz 1.3 -ph 2.0')
2. `dpdwetting brdgcline` : Contact line position (the height of each liquid slab, and the height of the pillars have to be given with '-dz 1.3 -ph 2.0')
3. `dpdwetting brdginterf` : 3D interface points at a given time
4. `dpdwetting brdgvelxz` : Velocity field on xy-plane
5. `dpdwetting brdgvelx` : Average x-velocity as a function of z-coordinate
6. `dpdwetting brdgzd` : Particle density as a function of z-coordinate
7. `dpdwetting brdgxzd` : Particle density map on xz-plane

2.4. Steady shear

Finally, one can perform the simulation of the equilibrated capillary bridge under steady shear. The final configuration after the equilibration (`2_equil_final.gro`) is used and the new control file (`3_shear.in`) includes information of the applied shear by `wallshear` and `wallshrgrp` arguments. The simulation can be done in the same way as before.

```
mpirun -np [number of cores] dpdwetting run -c 2_equil_final.gro -p
3_shear.in -o 3_shear
```

The modules `brdgpc` and `brdgsize` can be used to calculate the dynamic contact angle and the polymer concentration in the contact line region.

2.5. Simulation with different liquid-solid interaction strengths

The liquid-solid interaction strength is given in the topology file under the argument `[ Nonbonded ]`. As the liquid and solid particles are named as `W` and `S`, respectively, it has to be specified on the fifth column of the line for `S W`. It can be also specified in the first step of initial configuration generation, by giving the option `-sw` in the command `generate_polroughfilm.py`.

3. Reproduction of Paper Figures

Data for figures of the published papers can be reproduced by the following ways.

Fig 2(a)

Density profile of liquid particles can be obtained by `brdgzd` module with an equilibrium trajectory.

Fig 2(b)

Interface point can be calculated by the module `brdginterf` at a certain time with an equilibrium trajectory.

Figures 3, 4 and Table 2

Equilibrium contact angles at different liquid-solid interaction strength can be obtained by averaging contact angles in equilibrium by using `brdgsize` module.

Figures 5 and 7(c)

Dynamic contact angle at different shear velocities cna be obtained again by the `brdgsize` module with trajectories under shear.

Figure 6(a), 7(a), and 7(b)

Dynamic contact angle is obtained by the `brdgsize` module and the contact line position is calculated by the `brdgcline` module on the trajectory under shear.

Figure 6(b)

Density map of liquid particles can be obtained by the module `brdgxzd`.

Figure 8(a)

Velocity field can be calculated by the module `brdgvelxz`.

Figures 8(b) and 8(c)

Slip velocity can be obtained by the module `brdgvelx`.