

Molecular Dynamics simulations of shear droplets

Dataset documentation

Michele Pellegrino, Berk Hess

May 12, 2022

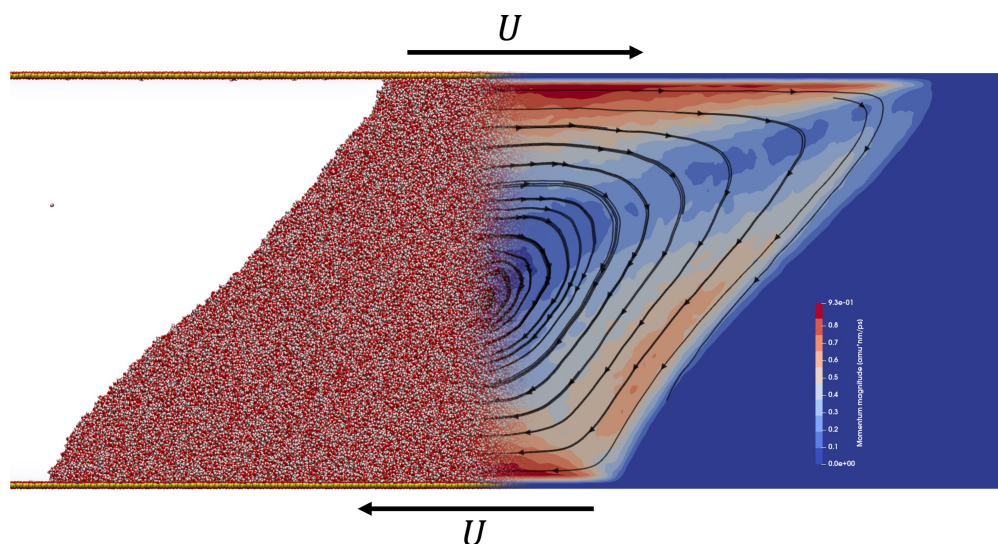


Figure 1: Depiction of the two faces of Molecular Dynamics outputs.

1 Introduction

This dataset contains the results of non-equilibrium Molecular Dynamic simulations of 2-dimensional SPC/E water nanodroplets confined between silica-like walls and under shear flow conditions, performed using Gromacs [1] [2]. The main purposes of these simulations are:

- To study the **motion of three-phases contact lines** over high-friction surfaces;
- To study the critical transition leading to **droplet breakage**;
- To test the **modelling and prediction** capabilities of continuous fluid dynamics simulation methods.

The results of the investigation of the points above is illustrated in an article, which has been published digitally and made available under open access on the Journal of Fluid Mechanics [3]; please refer to the paper for a detailed description of the molecular simulations and of the tested CFD methods. The publication of this dataset not only grants the reproducibility of the results discussed in the article, but also serves as collection of benchmarks for the fellow researchers willing to test improved and/or alternative models to describe the motion of contact lines.

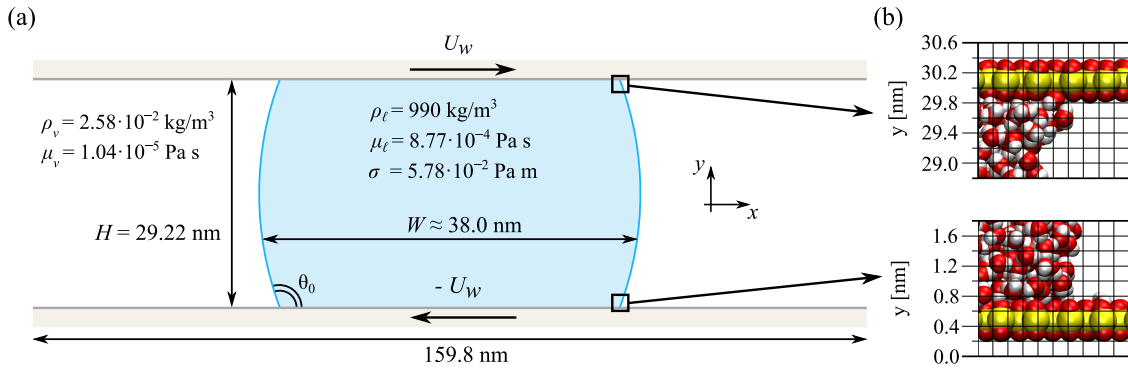


Figure 2: Schematic description of the shear droplet configuration (figure taken from [3]). (a) Geometry of the molecular system; here H refers to the wall separation distance, not to the MD box dimension along z , which is $Lz = 30.36600\text{nm}$. ρ : density, μ : viscosity, σ : surface tension. (b) Depiction of the grid of bins used to store fluid flow variables; $h_x \simeq 0.20\text{nm}$, $h_z \simeq 0.20\text{nm}$.

The output of molecular simulations have a ‘dual’ nature: trajectories and flow maps. Molecular trajectories depicts the system in full atomistic detail: this kind of output can become particularly heavy when written very frequently, especially for a system comprising of half a million atoms. The purpose of this study prevents the sacrifice of temporal resolution. Moreover, fully-detailed molecular information is not directly needed when comparing to CFD. Hence, a procedure to bin flow field variables on the fly was developed [4], reducing considerably the amount of storage needed and speeding-up analysis. This dataset mainly contains already-binned maps of flow variables. Nevertheless, the necessary configuration scripts needed to reproduce molecular trajectories with Gromacs are provided.

2 Structure of the dataset

📁 Molecular Dynamics simulations of shear droplets

📁 molecular-dynamics

📁 Q2Ca010

```
ener.edr
grompp.mdp
state.cpt
system.tpr
traj.xtc
```

📁 SilicaCharges

```
notes.txt
silica_cosres.itp
silica_q1.itp
```

...

```
atomtypes.itp
equilibrated_conf.gro
restraints_lambda0.gro
restraints_lambda1.gro
topology.top
```

📁 scripts

```
example_2020.dat
example_2021.dat
example.ipynb
readmap.py
```

documentation.pdf

q1-ca0000-2021.tar.gz

...

```
<charge>-<capillary no.>-<gmx version>.tar.gz
...
q4-ca0030-2020.tar.gz
MDpostProc.tar.gz
```

Flow field maps are set of *binary* .dat files, stored in tarballs which can be easily un-tarred by running:

```
tar -xvf <archive-name>.tar.gz
```

The folder ‘molecular-dynamics’ contains the configuration files needed to reproduce fully-detailed molecular trajectories. The ‘scripts’ folder contains examples of Python3 scripts that can be used to read the flow field data. Details are illustrated in the following sections. The archive `MDpostProc.tar.gz` contains the Python scripts used to process and analyze MD maps in [3]. It contains a self-explanatory README file.

3 Reading and analyzing flow field maps

Flow field data are produced using an ad-hoc Gromacs version [4]. Hydrodynamic variables are binned and stored on a uniform 2-dimensional grid. Being N_x the number of bins in the horizontal direction and N_z the number of bins in the vertical direction¹, one has for $i = 0, \dots, (N_x - 1)$ and $j = 0, \dots, (N_z - 1)$:

- **density** $\rho[i, j]$, expressed either in Dalton or Dalton/nanometer³;
- **velocity** $(v_x[i, j], v_z[i, j])$, expressed in nanometers/picosecond;
- **temperature** $T[i, j]$, expressed in Kelvin.

A Python3 code to read the .dat files and a Jupyter notebook illustrating a simple example can be found in the ‘scripts’ folder. Only `numpy` and `matplotlib` are required to run the scripts.

The Molecular Dynamics simulations have been mainly performed using Gromacs 2020; however, some simulations have been run with Gromacs 2021. It is important to distinguish between the two, not only in order to reproduce the results accordingly, but also because the custom Gromacs version that implements on-the-fly binning has evolved too; passing from 2020 to 2021 three main differences needs to be noted:

- The weight of .dat files has been greatly reduced (to $\sim 1/3$) and reading function made faster;
- A header with essential information has been added to each flow frame;
- The unit for density has moved from Dalton to Dalton/nanometer³ (i.e. now it’s a proper density);
- An artifact producing nonphysical velocity readings close to interfaces has been corrected.

It is therefore important to know with which version a flow frame has been generated in order to read it and analyze it accordingly. To the user willing to reproduce the simulations themselves we suggest to refer to the most recent version of the binning code [4].

Regarding files naming, the tarballs name has the following pattern:

```
qx-ca%4d-vers.tar.gz ,
```

where `qx` corresponds to the surface wettability (see `molecular-dynamics/SilicaCharges/notes.txt`), the integer after `ca` corresponds to 1000 times the Capillary number (see [3]) and `vers` corresponds to the Gromacs version used to generate the data. The frames naming goes as follow:

```
flow_%5d.dat ,
```

where the integer corresponds to the consecutive frame number. Frames have been sampled every 12.5ps.

¹With respect to [3], here the vertical coordinate is indicated with z instead of y . This is just due to convenience: in Molecular Dynamics no system is truly 2-dimensional, and it was easier to reserve y coordinate for the in-plane direction because of Gromacs-specific implementation of boundary conditions.

4 Reproducing molecular dynamics simulations

Examples of configuration files used to produce the molecular simulations are provided in the ‘molecular-dynamics’ folder. Below is a brief description of each type of file:

- `.itp` files include information regarding the force field and the topology of the system. In this project they are mainly used to describe the silica quadrupoles molecules that form the solid substrate.
- `.top` files are the ones to be passed to Gromacs’ pre-processor `gmx grompp`; they link to `.itp` files and provide additional information on the topology (e.g. total number of molecules for each group).
- `.gro` files represent molecular configurations to be passed to `gmx grompp` either as initial condition or to prescribe position restraints. They can be readily visualized via VMD [5].
- `.mdp` files contain all the numerical parameters of the molecular dynamics simulation (time step, the type of thermostat/barostat, etc...). See ‘Molecular dynamics parameters’ in [2].
- `.tpr` are binary topology files produced by `gmx grompp` and contain all the information on initial conditions, forces, position restraints and numerical parameters. They are the input of `gmx mdrun`, which effectively executes the simulation.
- `.cpt` are binary checkpoint files dumped by Gromacs throughout the simulation. They are typically used to re-start the simulation.
- `.xtc` files are one type of binary trajectory files.
- `.edr` files are used to store information about observables (energy, temperature, pressure, etc...); can be analyzed with `gmx energy`.

The wettability of the surface can be modified by changing the file read by the following `include` directive in `topology.top`:

```
#include "SilicaCharges/silica_qx.itp" ,
```

where each value of ‘x’ is mapped to a value for the equilibrium contact angle (see `SilicaCharges/notes.txt`). The wall velocity can be prescribed by tuning `delta-lambda` in `grompp.mdp`; to run an equilibrium simulation just specify `free-energy=no`. Further details can be found in [3], appendix C. The user-defined parameter to specify the number of bins and the output of flow frames can also be specified in `grompp.mdp`, see the comment on the configuration file `grompp.mdp` for further instructions.

The folder ‘Q2Ca0100’ contains an example of trajectory obtained from the simulation having capillary number $Ca = 0.1$ and equilibrium contact angle $\theta_0 = 95^\circ$. The steps to reproduce the simulation with Gromacs 2020 or 2021 are briefly illustrated below:

1. Run Gromacs’ pre-processor to obtain the input for `mdrun`:

```
gmx grompp -maxwarn 2 -c equilibrated_conf.gro -r restraints_lambda0.gro
↳ -rb restraints_lambda1.gro -p topology.top -f grompp.mdp -o Q2Ca0100/system.tpr
```

It is important to include the option `-maxwarn 2`, otherwise Gromacs will complain about the number of molecules in the position restraints configurations (`-r`, `-rb`) being different than the one in the initial configuration (`-c`).

2. Start the simulation by running:

```
cd Q2Ca0100
gmx mdrun -v -s system.tpr -flow
```

The `-flow` option tells `mdrun` to perform flow field binning on-the-fly. We refer to [2] for additional information on the options of `gmx mdrun`.

Given the size of the system it is highly recommended to run the simulations on in parallel on a cluster or a supercomputer. Getting high performance for this system is far from trivial due to its unusual geometry and due to the fact that a large part of the simulation box is basically empty; some indications on how to optimize `mdrun` in parallel can be found here: [6]. Finally, we highly recommend to run in double precision.

Versioning

The version of this database `x.y.z` is updated as follow:

- `x`: +1 when a new batch of flow maps is added
- `y`: +1 when the example configurations or the reading script are updated
- `z`: +1 when the documentation is updated

Acknowledgments

We acknowledge funding from Swedish Research Council (INTERFACE centre and grant nr. VR-2014-5680). We thank all the co-authors of [3], Dr. Uģis Lācis, Johan Sundin, Prof. Gustav Amberg, Prof. Stéphane Zaleski and Prof. Shervin Bagheri for the fruitful discussion on data management. We also thank Dr. Petter Johansson for developing the code to perform data collection. Numerical simulations were performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at PDC.

References

- [1] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*, SoftwareX vol. 1-2, 19-25 (2015)
- [2] Gromacs 2021 version, documentation: <https://manual.gromacs.org/documentation/2021/index.html>
- [3] U. Lācis, M. Pellegrino, J. Sundin, G. Amberg, S. Zaleski, B. Hess and S. Bagheri, *Nanoscale sheared droplet: Volume-of-Fluid, phase-field and no-slip molecular dynamics*, Journal of Fluid Mechanics, Journal of Fluid Mechanics, vol. 940 (2022), doi:10.1017/jfm.2022.219.
- [4] Fork of Gromacs which modifies it to enable output of two-dimensional flow fields, collected during a simulation: <https://github.com/pjohansson/gromacs-flow-field>
- [5] Visual Molecular Dynamics: <https://www.ks.uiuc.edu/Research/vmd/>
- [6] Getting good performance from `mdrun`, <https://manual.gromacs.org/documentation/2021/user-guide/mdrun-performance.html>