User Guide – SSA-tDPD module (v1.0)

1 Installation process

1.1 Installing Git

- In a terminal, type:

  ```
  sudo apt-get install git
  ```

  This will install git in your system. You have to create an account on Bitbucket (`https://bitbucket.org/`) to be able to access the ssa_tDPD repository.

- Now that Git is installed, at the first time you run any git command, the system will ask you a few things, so that the commit messages are correctly configured. Type:

  ```
  git config --global user.name "Your Name"
  ```

  ```
  git config --global user.email "youremail@domain.com"
  ```

1.2 Installing VTK

- In a terminal, type:

  ```
  sudo apt-get install vtk
  ```

  This will install the VTK library in your system.

1.3 Installing Paraview

- In a terminal, type:

  ```
  sudo apt-get install paraview
  ```

  This will install Paraview in your system. Paraview is a user-friendly VTK reader, used for post-processing Lammps results.

1.4 Optional: Installing MPI

- In a terminal, type:

  ```
  sudo apt-get install libopenmpi-dev openmpi-bin
  ```

  This will install OpenMPI in your system. It allows running Lammps using more than one processor.

1.5 Installing Lammps

- In a terminal, type:

  ```
  git clone git@bitbucket.org:bdrawert/ssa_tdpd.git
  ```

  to clone the repository from Bitbucket.

- Open the cloned folder, and navigate to the ./lib/vtk directory

  ```
  cd ssa_tdpd/lib/vtk
  ```

- Edit the file Makefile.lammps accordingly:

  ```
  vtk_SYSINC = -I/usr/include/vtk-X.X
  vtk_SYSLIB = -lvtkCommonCore -lvtkIOCore -lvtkCommonDataModel
  -lvtkIOXML -lvtkIOLegacy -lvtkIOParallelXML
  vtk_SYSPATH = -L/usr/lib64/vtk
  ```

  where X.X is the version of the VTK library installed in your system.

2 Compiling Lammps

- Open the ssa_tdpd folder, and navigate to the ./src directory. Then type:

  ```
  make yes-USER-VTK
  ```

  This will install the VTK package in Lammmps.

- Still in the /src directory, type:

  ```
  make mpi
  ```

  if MPI is installed. Alternatively, type `make serial` to install the serial version.

- If no error is displayed, the C++ compiler will generate a binary named `lmp_mpi` (or `lmp_serial`, if the case).

3 Running cases

- To run cases in parallel (MPI), use the following syntax:

  ```
  mpirun -np X path_to_lmp_mpi -in path_to_input_file
  ```

  where `X` is the number of MPI processes desired (for optimal performance, this number must match the number of physical processors in your machine). Note that `path_to_lmp_mpi` must be replaced by the path where the `lmp_mpi` file is located, and `path_to_input_file` must be replaced by the path of the input file.

- Alternatively, to run cases in serial, use the following syntax:

  ```
  ./path_to_lmp_mpi -in path_to_input_file
  ```

4 Input file structure

Example: flow past sphere

```
# DPD flow past a sphere (with diffusion of one species)

####################################
# Lammps setup
####################################
dimension 2 #enforces 2D simulation
units lj #reduced Lennard-Jones units
comm_modify mode single vel yes
atom_style ssa_tdpd/atomic #type of atom
neighbor 0.3 bin #creates neighbor list
neigh_modify delay 0 every 1 check yes


####################################
# Temporal integration setup
####################################
variable dt equal 0.0001 #time step
variable nt equal 15000 #number of time steps
variable freq_lagrangian equal 100 #freq. writing results (file)
variable freq_screen equal 100 #freq. writing results (screen)

####################################
# Domain setup
####################################
boundary p f p #(x,y,z); p = periodic, f = fixed
variable xmin equal -200 # minimum value of x
variable xmax equal 200 # maximum value of x
variable ymin equal -50 # minimum value of y
variable ymax equal 50 # maximum value of y
variable zmin equal -0.01 # minimum value of z
variable zmax equal 0.01 # maximum value of z
variable radius_external equal 2.2 # external radius
variable radius_internal equal 2.0 # internal radius
region domain1 block ${xmin} ${xmax} ${ymin} ${ymax} ${zmin}
   ${zmax} units box
region  sphere_region sphere 0 0 0 ${radius_external} units box
region domain union 2 domain1 sphere_region
create_box 2 domain #(Number of types ; region)
```

```
####################################
# Creates atoms and regions
####################################
lattice sq 1 #(lattice topology; number density)
create_atoms 1 region domain1 #(type; region region-ID)
create_atoms 2 random 2000 58131124 sphere_region #(type; random #
   particles; seed; region-ID)
mass * 0.01 #mass of particles
newton on
pair_style ssa_tdpd 1.58 354655456 #DPD parameters (rc, seed)
#      i  j  a     ga   si   s1    rc    rcc   kC  k0     s2
pair_coeff * *  1000   4.5  3.0  0.41  1.58  1.58  0  100.0  2.0

####################################
# Defines sphere region
####################################
# Removes a spherical region
group sphere region sphere_region #defines a group of atoms
region void1 sphere 0 0 0 ${radius_internal} units box #creates
   region
delete_atoms region void1 #deletes atoms within a region
group flow subtract all sphere #defines a group of atoms
velocity sphere set 0.0 0.0 0.0
fix  spherewall flow indent 10 sphere 0 0 0 ${radius_external}
   units box
set    group sphere type 2

####################################
# Initial velocity and concentration fields
####################################
# Velocity
variable U0 equal 208.0 #(label, initial velocity)
velocity all set ${U0} 0.0 0.0 #(group-ID, set vx vy vz)
# Concentration
set   group sphere ssa_tdpd/C 0.0 #(group group-ID, style, value)

####################################
# Info on screen
####################################
thermo         ${freq_screen}
thermo_style   custom step temp press pe ke
```

```
####################################
# Integration of particles' position, velocity, concentration
####################################
fix  pos_vel flow nve #(label, group-ID,style)
fix  conc all ssa_tdpd_verlet #(label, group-ID,style)

####################################
# Forcing zone
####################################
fix  forcing1 all ssa_tdpd/forcing 1 0 rectangle -195 25 5.0 25 1.0
    #(label, group-ID, style, frequency, species rank, geometry,
    centerX, centerY, length, width, value)
fix  forcing2 all ssa_tdpd/forcing 1 0 rectangle -195 -25 5.0 25
    0.0 #(label, group-ID, style, frequency, species rank,
    geometry, centerX, centerY, length, width, value)

####################################
# Enforce boundary conditions
####################################
fix  BCy all ssa_tdpd/reflect 1 ${ymin} ${ymax} #(label, group-ID,
    style, species, rank coordinate (x=0, y=1, z=2), coord_min,
    coord_max)

####################################
# Output results
####################################
dump  dmpvtk all ssa_tdpd/vtk ${freq_lagrangian} dump*.vtk id type
    vx vy vz C1 #(label, group-ID, style, frequency, filenames,
    variables to print)

####################################
# Run simulation
####################################
timestep ${dt}
run  ${nt}
```