User guide for CFD–DEM solver lammpsFoam

Rui Sun*[1], Prashant Gupta[2], Heng Xiao†[1] and Jin Sun[2]

[1]Department of Aerospace and Ocean Engineering, Virginia Tech, United States
[2]Institute for Infrastructure and Environment, The University of Edinburgh, UK

1 Introduction

Here is the brief introduction of demFoam, a toolbox of CFD–DEM simulation:

"demFoam" is the name of this toolbox.

"lammpsFoam" is an open source CFD–DEM solver combining the LAMMPS and openFoam.

"LAMMPS" is an open source molecular dynamics code that perform DEM simulations.

"OpenFOAM" is an open source toolbox for CFD simulations.

The structure of the demFoam is like this:

```
demFoam
|----Allwclean.sh
|----Allwmake.sh
|----cases/
|----documentation/
|----interfaceToLammps/
|----lammpsFoam/
|----readme/
```

"Allwclean.sh" and "Allwmake.sh" are written to help with the compilation of lammpsFoam, the CFD–DEM solver in demFoam. "Allwmake.sh" will help you install lammpsFoam while "Allclean.sh" will uninstall lammpsFoam. Note that the source file of LAMMPS is not included in demFoam. We highly recommend you use lammps-1Feb14, a version released on February 2014.

All numerical cases are placed in "demFoam/cases/", including verification cases and validations cases.

Some documents are in "demFoam/documentation/", with a basic user manual (userManual.tex).

"demFoam/interfaceToLammps/" and "demFoam/lammpsFoam/" are folders containing the source code of lammpsFoam. To compile lammpsFoam, the files of "demFoam/interfaceToLammps/" should be copied to "LAMMPS-(some version)/src". Allwclean.sh are Allwmake.sh written to help with the compilation of lammpsFoam, the CFD–DEM solver

How to install lammpsFoam on your linux machine?

Although Allwmake.sh can automatically help you with the installation of lammpsFoam, it is necessary to demonstrate individual steps in the compilation.

*Email: sunrui@vt.edu
†Email: hengxiao@vt.edu

Environment for lammpsFoam The CFD-DEM solver lammpsFoam would be successfully installed on a machine with OpenFOAM and LAMMPS. Here are the steps to compile it on your linux machine:

1. Install OpenFOAM on your machine, the up-to-date lammpsFoam is supported for OpenFOAM 2.1.1.
2. Copy all the files in "InterfaceToLammps" folder to "lammps-1Feb14/src", or create a soft link of the files.
3. Change to "lammps-1Feb14/src/STUBS". Use command "make".
4. Change to the "lammps-1Feb14/src" folder. Use command "make yes-granular" to install the granular package of LAMMPS. Add "kspace", "manybody", "molecule", "fld" as well in a similar way.
5. Stay at the "lammps-1Feb14/src" folder. Use command "make shanghailinux" to generate an executable file named "lmp_shanghailinux" in this folder. It may take a while.
6. Stay at "lammps-1Feb14/src" folder. Use command "make makeshlib" to generate a new "Makefile.shlib" file.
7. Stay at "lammps-1Feb14/src" folder. Use command "make -f Makefile.shlib shanghailinux" to compile LAMMPS to a dynamic library. After this step, you will see a file named "liblammps_shanghailinux.so".
8. Change to "$FOAM_USER_LIBBIN" and make a soft link of "liblammps_shanghailinux.so" in this folder. "liblammps_shanghailinux.so" is the file generated in the previous step.
9. Change to "LammpsFoam" folder. Use command "wmake libso dragModels" to compile the drag models. Use command "wmake libso chPressureGrad" to compile the pressure gradient for channel flow.
10. Stay at the same directory. Use command "wmake" to compile lammpsFoam.

To check if your lammpsFoam is succesfully compiled, you can:

1. Change to the "cases/auto-testing/test-cases/xiaocase3/" folder.
2. Execute "./Allrun.sh" in the command line to see the result.

2 Cases and supporting publications

There are lot of publications using lammpsFoam [1, 2, 3, 4, 4, 5, 6, 7, 8, 9]:
Several cases from the publications are added in "demfoam/cases/auto-testing/test-cases":

1. "xiaocase1" is the 1st case in H. Xiao et al.'s paper [2].
2. "xiaocase3" is the 3rd case in H. Xiao et al.'s paper [2].
3. "expMuller09" is the case in R. Sun et al.'s paper [8, 9].

References

[1] J. Sun, H. Xiao, and D. Gao. Numerical study of segregation using multiscale models. *International Journal of Computational Fluid Dynamics*, 23:81–92, 2009.

[2] H. Xiao and J. Sun. Algorithms in a robust hybrid CFD–DEM solver for particle-laden flows. *Communications in Computational Physics*, 9:297–323, 2011.

[3] P. Gupta, J. Sun, and J. Ooi. Segregation studies in bidisperse fluidsed beds: CFD–DEM approach with DNS based drag force models. In *UK China Particle Technology Forum, Birmingham*, July 2011.

[4] P. Gupta, J. Sun, and J. Ooi. Study of drag force models in simulation of bidisperse gas–solid fluidised beds using a CFD–DEM approach. In *The 8th European Congress of Chemical Engineering, Berlin*, September 2011.

[5] P. Gupta, H. Xiao, and J. Sun. Effect of drag force models on simulation of segregation in gas–solid fluidised beds using a CFD–DEM approach. In *UK–China Particle Technology Forum*, July 2011.

[6] P. Gupta, M. Ebrahimi, M. Robinson, M. Crapper, J. Sun, M. Ramaioli, S. Luding, and J.Y. Ooi. Comparison of coupled CFD–DEM and SPH–DEM methods in single and multiple particle sedimentation test cases. In *PARTICLES*, Stuttgart, Germany, September 2013.

[7] P. Gupta, J. Sun, and J. Ooi. Effect of hydrodynamic interactions on the segregation rate in bi-disperse gas–solid fluidised bed and validation studies. In *7th International Conference for Conveying and Handling of Particulate Solids – CHoPS*, September 2012.

[8] R. Sun and H. Xiao. Diffusion-based coarse graining in hybrid continuum–discrete solvers: Theoretical formulation and a priori tests. Submitted, 2014.

[9] R. Sun and H. Xiao. Diffusion-based coarse graining in hybrid continuum–discrete solvers: Application in CFD–DEM solvers for particle laden flows. Submitted, 2014.