Trials of an Open Source DSMC code

Tom Scanlon[1], Ehsan Roohi[2], Graham Macpherson[3] and Jason Reese[1]

[1] University of Strathclyde, Department of Mechanical Engineering, Glasgow G1 1XJ, Scotland
[2] Sharif University of Technology, Department of Aerospace Engineering, Tehran, Iran
[3] OpenCFD Ltd., 9 Albert Road, Caversham, Reading, RG4 7AN

This paper presents the results of DSMC simulations using dsmcFoam – a recently developed module of OpenFOAM. OpenFOAM (Open Field Operation and Manipulation) is a programmable CFD toolkit supplied with source code and compilers. OpenFOAM is produced by OpenCFD Ltd., is freely available and open source and licensed under the GNU General Public Licence.

Customised applications are created for specific problems, e.g. DSMC, using the functionality built into generic OpenFOAM libraries (modules). The DSMC code has been written to take advantage of the flexibilities associated with the OpenFOAM C++ structure and hierarchical design. The open source nature of the source code means that its solvers, utilities and libraries are fully extensible. The use of advanced level C++ as the core programming language brings major benefits to users:

- Advanced error checking at both complile and run time.
- Extremely robust solver and utility executables.
- High speed calculation with efficient memory management and fast linear equation solvers.
- Parallel processing with linear speed up with number of processors.

The core technology of OpenFOAM is a flexible set of efficient C++ modules. These are used to build a wide variety of solvers, to simulate specific problems in engineering mechanics; utilities, to perform pre- and post-processing tasks ranging from simple data manipulations to visualisation and mesh processing and libraries. This creates toolboxes that are accessible to the solvers/utilities.

Domain decomposition parallelism is fundamental to the design of OpenFOAM and integrated at a low level so that solvers can generally be developed without the need for any ’parallel-specific’ coding. dsmcFoam has been constructed to take advantage of this parallel capability and the most complex functionality within it is the parallelised particle tracking feature in polyhedral meshes[1].

The dsmcFoam code has been developed to include the following features:

- Steady / transient solutions.
- Parallelised code.
- Arbitrary 2D/3D geometries.
- Automatic generation of sub-cells.
- Multiple species, real gas mixtures.
- Boundary conditions including particle inlet/outlet, Maxwellian diffusive and specular walls and freestream boundaries.
- Elastic/inelastic collision models (VHS, Borknakke-Larsen)

The initial test results for collisions rates, 2D and 3D cases compare well with the benchmark analyses of Bird's code[2].

Future work will include low-Ma pressure driven channel flows (IP), incorporation of more sophisticated collision models (VSS, GHS), a continuum CFD-DSMC hybrid code and chemically reacting flows.

References

[1] Graham B. Macpherson, Niklas Nordin and Henry G. Weller Particle tracking in unstructured, arbitrary polyhedral meshes for use in CFD and molecular dynamics, *Communications in Numerical Methods in Engineering* vol. 25, no. 3, pp. 263-273, 2009 http://dx.doi.org/10.1002/cnm.1128

[2] Bird, G.A. *Molecular Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, 1994.