Proceedings of ASME ICNMM2008
Sixth International Conference on Nanochannels, Microchannels, and Minichannels
June 23–25, 2008, Darmstadt, Germany

DRAFT — ICNMM2008-62277

MOLECULAR DYNAMICS FOR FLUID MECHANICS IN ARBITRARY GEOMETRIES

Graham B. Macpherson and Jason M. Reese
Department of Mechanical Engineering,
University of Strathclyde, Glasgow G1 1XJ, UK
graham.macpherson@strath.ac.uk
jason.reese@strath.ac.uk

ABSTRACT

Simulations of nanoscale systems where fluid mechanics plays an important role are required to help design and understand nano-devices and biological systems. A simulation method which hybridises molecular dynamics (MD) and continuum computational fluid dynamics (CFD) models is able to accurately represent the relevant physical phenomena and be computationally tractable.

An MD code has been written to perform MD simulations in systems where the geometry is described by a mesh of unstructured arbitrary polyhedral cells that have been spatially decomposed into irregular portions for parallel processing. The MD code that has been developed may be used for simulations on its own, or may serve as the MD component of a hybrid method. The code has been implemented using OpenFOAM, an open source C++ CFD toolbox (www.openfoam.org).

The requirements for two key enabling components are described. 1) Parallel generation of initial configurations of molecules in arbitrary geometries. 2) Calculation of intermolecular pair forces, including between molecules that lie on mesh portions assigned to different, and possibly non-neighbouring processors.

A case study of flow in a realistic nanoscale mixing channel, where the geometry is drawn and meshed in engineering CAD tools is simulated to demonstrate the capabilities of the code.

INTRODUCTION

Simulations of nano scale liquid systems can provide insight into many naturally-occurring phenomena, such as the action of proteins that mediate water transport across biological cell membranes [1]. They may also facilitate the design of future nano devices and materials (e.g. high-throughput, highly selective filters or lab-on-a-chip components). The dynamics of these very small systems are dominated by surface interactions, due to their large surface area to volume ratios. Direct simulation of the fluid using molecular dynamics (MD) [2, 3] presents an opportunity to model these phenomena with minimal simplifying assumptions.

Successful fluid dynamics simulations using molecular dynamics have been reported [4–7], but MD is prohibitively computationally costly for simulations of systems beyond a few tens of nanometres in size, over timescales beyond a few tens of nanoseconds. Fortunately, the molecular detail of the full flowfield that MD simulations provide is often unnecessary; in liquids, beyond 5–10 molecular diameters ($\lesssim$ 3nm for water) from a solid surface the continuum-fluid approximation is valid and the Navier-Stokes equations with bulk fluid properties may be used [7–9]. Hybrid simulations have been proposed [10–14] to simultaneously take advantage of the accuracy and detail provided by MD in the regions that require it, and the computational speed of continuum mechanics in the regions where it is applicable.

MD IN ARBITRARY GEOMETRIES

Published studies have demonstrated that hybrid simulations are viable, but these studies have dealt only with simple flows and domains. The geometries used have been typically simple cuboids with periodic boundaries. This is primarily because existing, widely used MD codes (such as LAMMPS, NAMD, AMBER, GROMACS or DL_POLY) can only simulate systems rep-

resented by simple domains: volumes that are space filling when periodic boundaries are applied, normally cubes, cuboids, or parallelepipeds. This is because most MD simulations are intended to examine a system in an infinite, unbounded medium, without the influence of solid surfaces.

In order to produce a useful, general simulation tool for hybrid simulations, the MD component must be able to model complex geometrical domains represented by unstructured meshs of arbitrary polyhedra (such as those generated by large scale, automatic meshing of geometries created by engineering CAD tools) that have been spatially decomposed for distributed parallel computing. This is also useful functionality in its own right; MD simulations of complex nano-devices derived from CAD models can be made directly by performing 'CFD with molecules'. To achieve this:

- initial configurations of molecules corresponding to volumes defined by the mesh must be generated. The algorithms underpinning a preprocessing tool able to create such configurations are described in reference [15]. This is able to fill volumes defined by a zone of the mesh (a set of cells) with a single species crystal lattice of molecules. The user may specify the lattice structure, orientation, density, temperature and average velocity;
- intermolecular forces must be calculated, taking account of periodic and interprocessor boundaries. This is the most important and computationally demanding aspect of any MD simulation and methods are detailed in reference [16];
- molecules must be tracked as they move through the mesh from cell to cell [17];
- boundary conditions and constraints must be imposed to control the state and define the dynamics of the system.
- spatially resolved measurements of the properties of the fluid and the flow must be made.

Parallelisation

The simulation is to be parallelised by domain decomposition [2, 18, 19], where the simulation volume is divided into smaller portions and each processor is given responsibility for simulating the molecules residing in its portion. Molecules move between processors when they enter and leave a portion. Where the simulation domain is simple (a cube or a cuboid for example) then the decomposition is straightforward (divide into smaller cubes or cuboids), which is what is done by existing MD codes. Where the domain is a complex shape constructed using unstructured arbitrary polyhedra, the volume must be decomposed into irregular portions. There are libraries available for partitioning meshes which minimise interprocessor connections, but can produce unintuitively shaped portions over which the user does not have much direct control.

Intermolecular forces must be passed across processor boundaries, and where the decomposition into portions can be entirely arbitrary, processors that do not share a boundary still may need to communicate, the details of this are described in reference [16].

Implementation in OpenFOAM

All algorithms described have been implemented in OpenFOAM [20], an open source C++ library intended for continuum mechanics simulation of user-defined physics (primarily used for CFD) in arbitrary, unstructured geometries. The MD simulation code used here [16] has been built using OpenFOAM's lagrangian particle tracking library [17] and is called gnemdFOAM. All features of OpenFOAM have a common infrastructure for distributed memory parallel processing using MPI for communications.

The initial molecule configuration generation tool for gnemdFOAM is also written using OpenFOAM and is called molConfig [15]. molConfig operates independently on individual portions of a mesh that have been spatially decomposed to run in parallel, allowing systems comprising very large numbers of molecules to be created because they never need to all be contained in the memory of a single computer. The molecular configurations are the same whether generated in parallel or in serial; crystal lattices generated in parallel are continuous and defectless across interprocessor boundaries. All parallel decomposition and reconstruction is dealt with by OpenFOAM using existing functionality.

Mesh description

The mesh in OpenFOAM is flexible and powerful: it is unstructured and built from arbitrary polyhedra. From the OpenFOAM user guide [20]:

> "By default OpenFOAM defines a mesh of arbitrary polyhedral cells in 3D, bounded by arbitrary polygonal faces, i.e. the cells can have an unlimited number of faces where, for each face, there is no limit on the number of edges nor any restriction on its alignment."

A list of mesh vertex positions is stored and a list of mesh faces is constructed; each face is an ordered list of vertex numbers. Cells are constructed as a list of face numbers. Vertices may be shared by several faces and cells. Cells can be grouped together into zones, each zone representing a region of the domain with common characteristics. Zones are used in molConfig to define regions to be filled with different crystals.

CASE STUDY: CAD-DERIVED MIXING CHANNEL

To demonstrate the capabilities of gnemdFOAM, a case study of flow in a complex, 3D nanochannel was simulated where the geometry was derived from a CAD (computer-aided design)

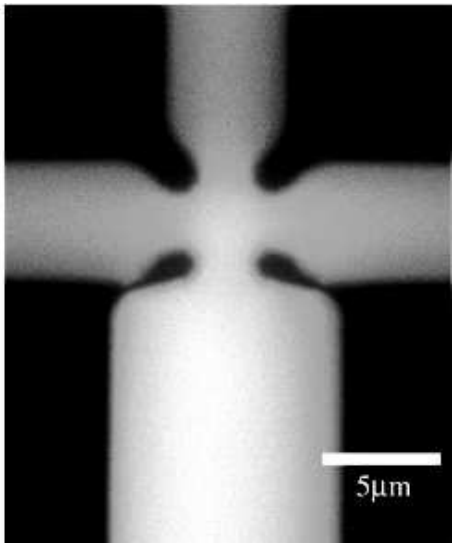

Figure 1. The geometry that the case study is based on, taken from [21]. There are three fluid inlets (narrower channels at the top of the image) and one outlet.

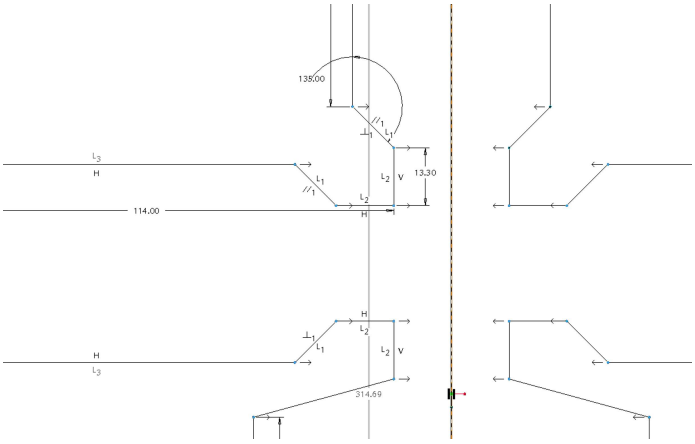

Figure 2. The Pro/ENGINEER® defining sketch of the mixing channel case study. Note that the right-hand-side is a mirrored copy of the left-hand-side, and that the mixing section dimensions are determined by parameter L_2. Therefore, changing this length (currently 13.3 reduced units) changes the whole mixing section as one entity.

model. The objective of the study was not primarily to analyse the fluid dynamics in the channel, rather to demonstrate what it is possible to simulate. A three inlet, one outlet microscale mixing channel from reference [21] was chosen as a guide for the geometry, see figure 1, and a reduced scale version was created.

Geometry Definition

The process of creating the geometry is:

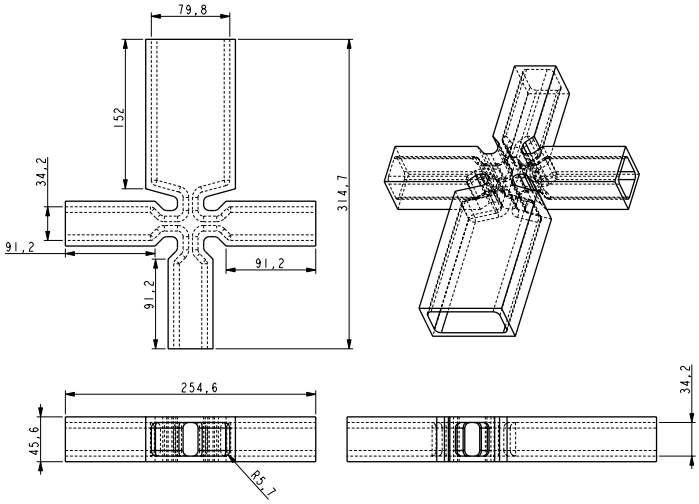

Figure 3. The overall dimensions, shown in reduced units where $\sigma_r = 0.34nm$, so the overall length of $314.7 = 107nm$.

1. The system was drawn in Pro/ENGINEER®, a commercial CAD tool. A sketch is created defining the outline of the shape, see figure 2. Note that constraints are placed on most of the dimensions so that they may only be one of a small set of parameters, allowing the size of the mixing section to be easily changed. Adjusting L_2 adjusts the width of the mixing section. All dimensions are in MD reduced units, where $\sigma_r = 0.34nm$. This sketch is extruded to form a solid, hollowed out to leave a shell and the material at the inlets and outlets removed, see figure 3. The internal edges and corners are all rounded to make the geometry more realistic, as fabrication techniques are not able to make sharp channel shapes. This gives the volume of the wall regions with empty space remaining for the fluid section.
2. The geometry is exported from Pro/ENGINEER® as a STEP file and imported into GAMBIT®, a commercial mesh generation tool.
3. A volume exists for the region that will form the solid walls and a region must be created for the volume containing the fluid. To do this faces for the inlets and outlets are created from existing edges.
4. These new faces are combined with the existing faces on the inside of the channel to create a volume for the fluid section. This is shown in figure 4 where the geometry involved only in the solid wall volume is not shown.
5. The fluid volume is split into 4 parts by creating box volumes surrounding each of the three inlets channels and intersecting them with the fluid volume.
6. A reservoir volume is created at the entrance to each of the inlet channels and joined to the adjacent inlet channel volume.
7. Each volume is given a name to identify all of the cells that

are created in it as being part of a zone for the generation of initial configurations of molecules.

8. Collections of geometrical faces are grouped together to form patches, so that all of the cell faces that lie on them will form patches in the mesh to which boundary conditions may be applied. The outlet, the faces of the molecule reservoirs that are external to the volumes and the remaining outer surface are defined as three separate patches.
9. The volumes are meshed by GAMBIT® using mostly automatic tetrahedral meshing, except in the molecule reservoirs where hexahedral cells were easy to create. A size function is applied at the mixing section to give finer cells around the mixing section. Creating meshes with tetrahedral cells is discouraged in continuum CFD, despite their ease of creation by automated tools, because they suffer from large interpolation errors compared to hexahedral cells. In this case the cells are required only to provide a space filling representation and to collect measurement data; no equations are solved using them. It will be seen, however, that collecting data in tetrahedral cells may produce noisy data.
10. The mesh is exported from GAMBIT® as a Fluent® mesh and imported into OpenFOAM using the fluentMeshToFoam utility with the 'writeZones' option. The mesh as imported is shown in figures 5, 6 and 7.
11. The boundary file in the OpenFOAM mesh directory is edited to assign the correct type to the patches created in the meshing process. The outlet patch and external surfaces except those of the molecule reservoirs are of type 'patch,' meaning that a molecule will be deleted when it touches a face. The reservoir external surfaces are of type 'wall,' and molecules are specularly reflected from faces they impact on. The external faces are of type 'patch' because the crystal near them is tethered forming a solid, so flow cannot leave the domain. The wall molecules whose tether points are close enough to the edge of the domain such that their locus of oscillation crosses the patch will be deleted. This does not affect the flow in the system if the wall is thick enough and is preferable to wall molecules close to the edge of the domain bouncing rapidly against a solid wall.

At this stage the geometry definition is complete.

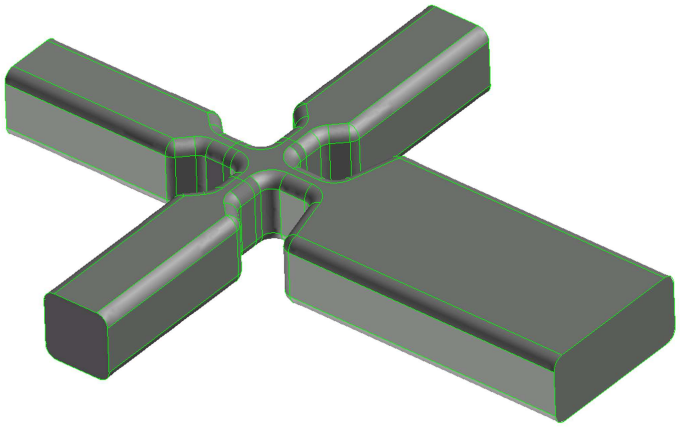

Figure 4. The newly created faces are combined with the faces on the inside of the existing volume to define an internal volume for the fluid. The geometry not associated with this new volume is not shown

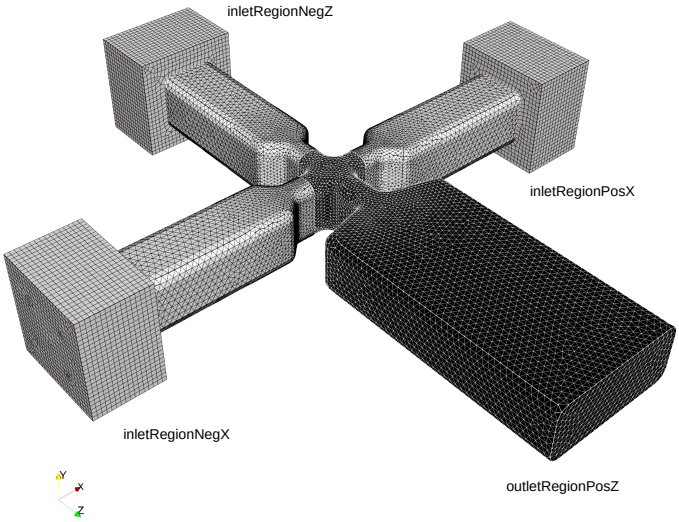

Figure 5. The mesh after being exported to OpenFOAM. The four liquid zones are shown.

MD Preprocessing and Simulation

Three types of molecule are created and are named depending where on where in the geometry they are initially placed. This allows the mixing in the channel to be observed by measuring the relative abundance of each type. There are two liquid molecule types, separated into those that start in the centre of the channel (the outlet region and the aligned inlet) and those that start in the side inlets. Molecules that comprise the wall are also created as a separate species to allow the channel boundary to be identified. The molecules have the ids LJ_C, LJ_S and LJ_W, meaning Lennard-Jones, Centre, Side and Wall)

Four simulations were carried out in this geometry at two initial temperatures, $T = 1.0$ in reduced units, equivalent to 120K, and $T = 2.5 \equiv 300K$. One simulation at each temperature used the same intermolecular potential between each molecule (shifted force Lennard-Jones, $\sigma = 0.34nm, \varepsilon = 120k_b$ [3]) and another simulation used potentials where the energy scale of the shifted force Lennard-Jones potential was altered to make the liquids less miscible. To do this the attractive well was made deeper between liquid molecules of the same type

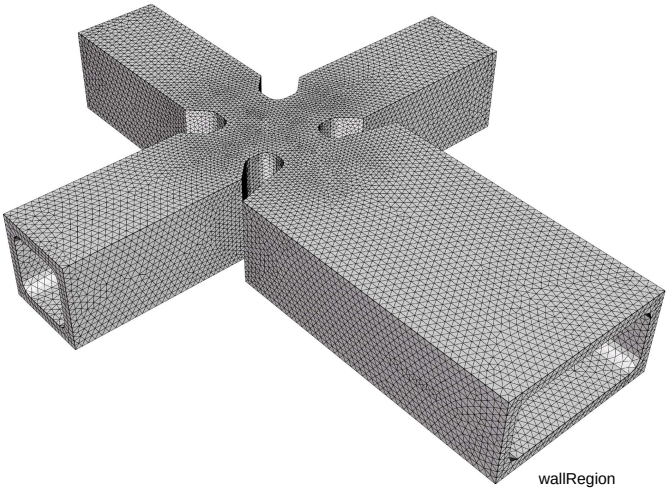

Figure 6. The mesh after being exported to OpenFOAM. The wall zone is shown.

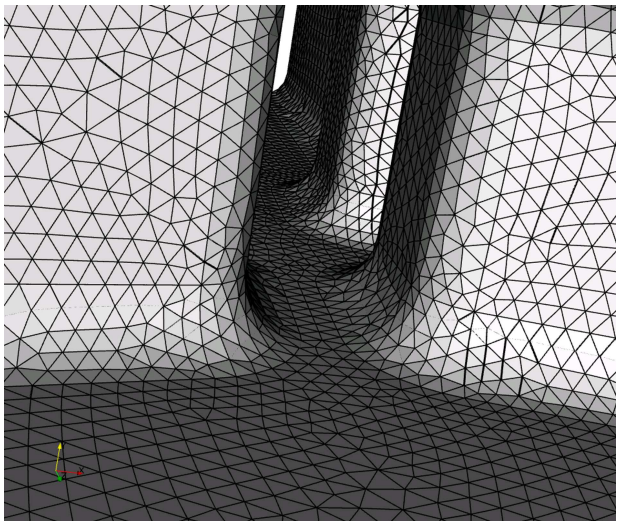

Figure 7. The internal space forming the liquid channel. Note the curved channel edges.

($\varepsilon = 1.25 \times 120k_b$) and shallower between molecules of different types ($\varepsilon = 0.75 \times 120k_b$). The initial configuration of the systems were otherwise identical. Wall molecules interact with liquid molecules and each other with a shifted force Lennard-Jones potential ($\sigma = 0.34nm, \varepsilon = 120k_b$) and are tethered into place by a harmonic spring potential. All molecules have a mass of $6.6904 \times 10^{-26}kg$. No thermostat is applied to the simulation.

The molecule reservoirs are enclosed by solid walls and flow is created in the system by creating an outlet that deletes molecules from the simulation when they cross it. This effectively means that the system vents into a vacuum, although the

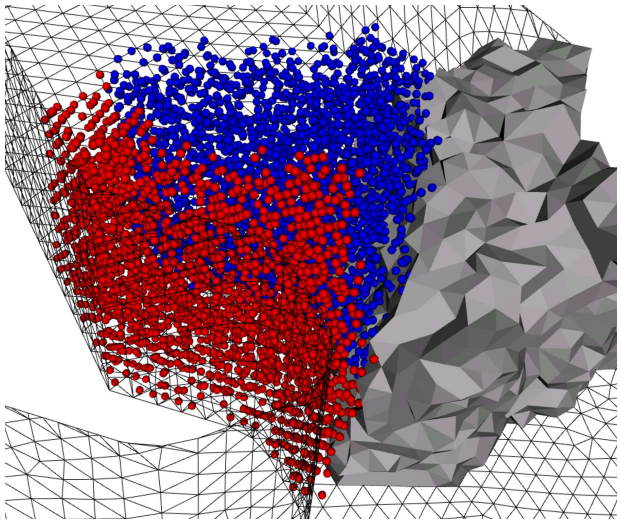

Figure 8. The molecules contained on one processor portion of mesh are shown coloured by id: blue for LJ_C and red for LJ_W. The mesh portion of an adjacent processor is also shown.

fact that a molecule is deleted as soon as it touches the patch means that it is not allowed to evaporate into the vacuum naturally: it is 'pulled' out of the simulation. The steps required to preprocess and run the MD simulation are:

1. The mesh is decomposed into 48 portions using the METIS library [22], which is used by OpenFOAM's decomposePar utility.
2. The domain is filled with molecules in parallel using molConfig. The liquid zones (LJ_C and LJ_S) are filled with a simple cubic lattice and the wall molecules (LJ_W) are tethered into an FCC lattice. Figure 8 shows the molecules that are created on one of the processors, and the mesh portion belonging to an adjacent processor.
3. The controlDict (control dictionary file, common to all OpenFOAM simulations) has the timestep set to 0.005, with a simulation end time of 500 (both in reduced units). An interval of 10 time units (2000 timesteps) is specified to write the configuration of the system to disk (from which it can be restarted if necessary) and acts as the averaging period for measuring the field values. The temperature, velocity, number density and mass density fields are measured for each species and as total values. The mass and mole fraction fields can be calculated during post-processing using the ratio of species to total density.

The molecule creation and simulation runs were carried out on 48 cores of a 100 core cluster, each core belonged to a dual core, 64bit, 2GHz AMD Opteron™ chip. The interconnect was via gigabit ethernet. It took approximately 13 minutes to create the initial configuration of 1462512 molecules for each case. The

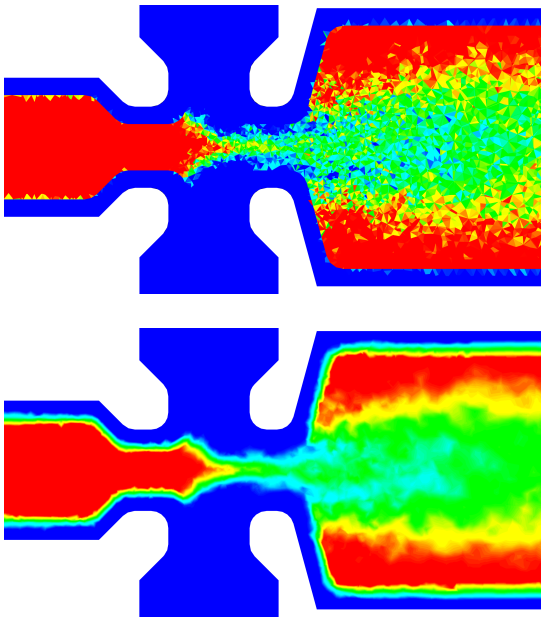

Figure 9. Comparing the raw cell data and the interpolated field. The raw cell data (above) is more difficult to appreciate in the mixing flow field but forms a sharp interface at the wall, the interpolated field (below) smears the interface at the wall but is easier to appreciate in the flow field.

number of molecules created in all simulations is the same because the density, orientation, structure and anchor of the lattices stays the same for each. The MD simulations took between 80 and 136 hours to run. Building all referred cells and interaction lists took approximately 11 minutes and 30183 liquid molecules were removed at zone boundaries due to high energy overlaps. The time taken to solve varies partly due to the differing rate of outflow of molecules from the system (faster flow leaving fewer molecules to simulate in later timesteps) and also due to disk access and communications contention with other jobs running on the cluster.

Results

Examples of the results from the simulations are shown in four figures. All figures were generated using [23]. Each figure shows the mole fraction of the side species (LJ_S) for the stated averaging period. The four figures are:

- figure 10: equal intermolecular potentials and $T = 2.5$ for the 210 timestamp;
- figure 11: different intermolecular potentials and $T = 2.5$ for the 210 timestamp;
- figure 12: equal intermolecular potentials and $T = 1.0$ for the 500 timestamp;
- figure 13: different intermolecular potentials and $T = 1.0$ for the 500 timestamp;

The timestamp for an averaging period refers to the data collected in the previous 10 units of time, so a timestamp of 210 represents the data collected between 200 and 210. The temperatures stated are the initial temperatures of the simulation, defined by molConfig. In each simulation the temperature drops because the system is depressurising and evaporating.

All results shown are from a cut through the middle of the domain along a plane with normal in the y direction (see figure 5 for the axes of the geometry). The data is collected in cells, but can be displayed as an interpolated field by ParaView. The two representations are compared in figure 9. The interpolated fields are necessary to calculate and display contours of the field variables.

There is a significant difference between the $T = 2.5$ and $T = 1.0$ simulations. In the $T = 2.5$ cases the outlet section rapidly depressurises and the flow 'chokes' at the throat of the mixer where very high velocities ($> 200 m/s$) are observed. The side and centre species mix in a complex process because the state at the throat is close to the critical point for a Lennard-Jones fluid with the energy and length scales used here [24]. Where the intermolecular potentials are different, the mixing of the two streams is not as complete; the side streams stay relatively sepa-

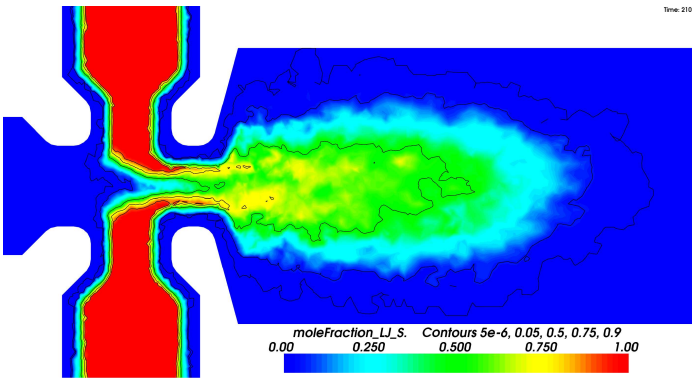

Figure 10. Equal potentials, $T = 2.5$, time = 210

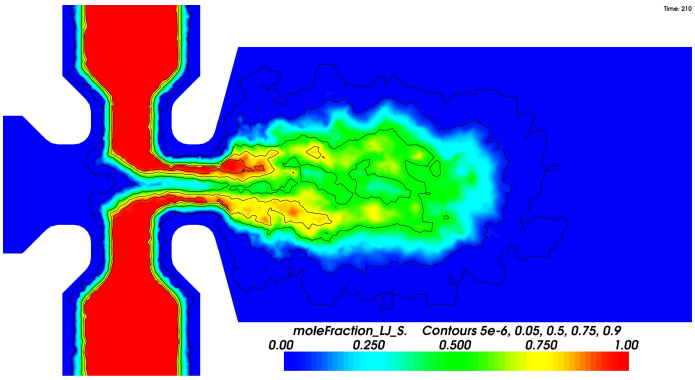

Figure 11. Different potentials, $T = 2.5$, time = 210

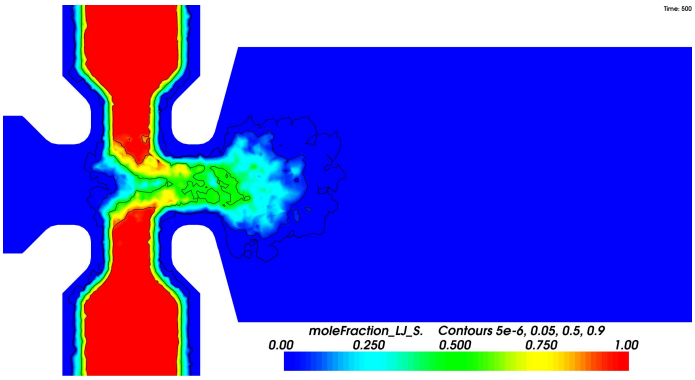

Figure 12. Equal potentials, $T = 1.0$, time = 500

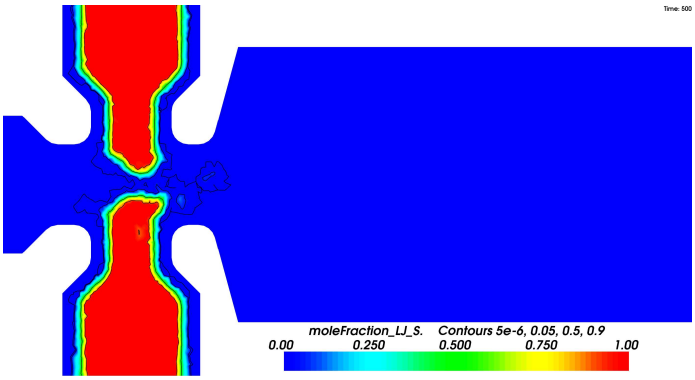

Figure 13. Different potentials, $T = 1.0$, time = 500

rate from the centre. The 5×10^{-6} mole fraction contour shows the extent to which traces of one species have diffused into the other. Where the potentials are equal, the species diffuse into each other more readily.

In the $T = 1.0$ cases the flow is significantly slower and does not undergo a large decompression in the outlet section. The equal potential fluids mix well and diffuse into each other, whereas in the different potential simulation the fluids stay relatively immiscible and do not diffuse into each other significantly. The different potential simulation does not decompress as much, or flow as fast as the equal potential simulation. The overall outflow can be appreciated by plotting the total number of molecules in the system as a function of time, see figure 14.

DISCUSSION AND CONCLUSION

A case study has been presented showing a molecular dynamics simulation of the mixing of liquids in a complex nanoscale geometry, derived from a CAD model. The simulations were performed to test and demonstrate the capabilities of a new MD code (gnemdFOAM [15, 16]) that has been developed to perform simulations in geometries defined by meshes of unstructured arbitrary polyhedra. The code has been implemented in OpenFOAM [20], and open source C++ simulation toolbox, and may be used for simulations itself, or serve as the MD component of a hybrid MD/continuum simulation.

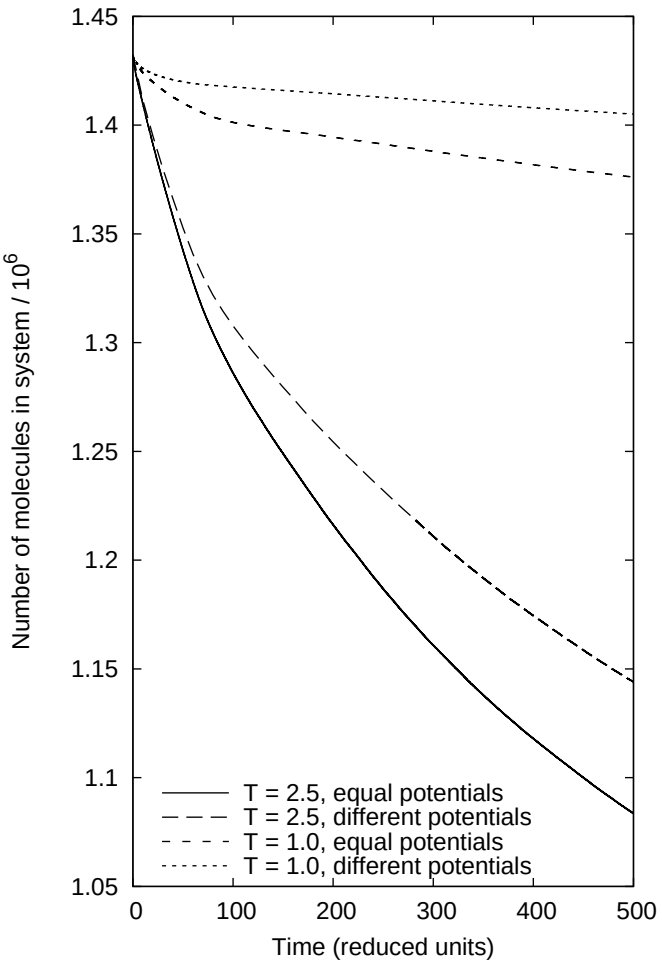

Figure 14. The total number of molecules in the system versus time. The initial decompression of the system can be seen in the steeper gradient sections. Reducing the temperature reduces the flowrate. The different intermolecular potential simulations flow more slowly than those where the potentials are equal.

In the case presented is very difficult to make reliable comparisons between the simulations because the fluid state is uncontrolled and the alteration of intermolecular potentials alters the properties and transport coefficients of the fluid. The use of an outlet venting to vacuum is also an uncontrolled way to drive the flow. The intention of this case study was to illustrate the potential of the code to perform simulations in complex geometries. It has achieved this aim, and highlighted the need for the infrastructure to be created to make such simulations more controllable.

Acknowledgements

The authors would like to thank Chris Greenshields and Matthew Borg of Strathclyde University, and Henry Weller and Mattijs Janssens of OpenCFD Ltd. for their help and useful discussions. This work was funded in the UK by the EPSRC (EP/F002467/1) and through a Philip Leverhulme Prize for JMR from the Leverhulme Trust.

REFERENCES

[1] Agre, P., 2006. "The aquaporin water channels". *Proceedings of the American Thoracic Society,* **3**(1), pp. 5–13.

[2] Rapaport, D. C., 2004. *The Art of Molecular Dynamics Simulation*, 2nd ed. Cambridge University Press.

[3] Allen, M., and Tildesley, D., 1987. *Computer Simulation of Liquids*. Oxford University Press.

[4] Hirshfeld, D., and Rapaport, D., 1998. "Molecular dynamics simulation of Taylor-Couette vortex formation". *Physical Review Letters,* **80**(24), pp. 5337–5340.

[5] Rapaport, D., 2006. "Hexagonal convection patterns in atomistically simulated fluids". *Physical Review E,* **73**(2), p. 25301.

[6] Travis, K., Todd, B., and Evans, D., 1997. "Poiseuille flow of molecular fluids". *Physica A,* **240**(1-2), pp. 315–27.

[7] Okumura, H., and Heyes, D. M., 2004. "Comparisons between molecular dynamics and hydrodynamics treatment of nonstationary thermal processes in a liquid". *Physical Review E,* **70**(6), p. 061206.

[8] Koplik, J., and Banavar, J. R., 1995. "Continuum deductions from molecular hydrodynamics". *Annual Review of Fluid Mechanics,* **27**, pp. 257–292.

[9] Becker, T., and Mugele, F., 2003. "Nanofluidics: viscous dissipation in layered liquid films". *Physical Review Letters,* **91**(16), p. 166104.

[10] O'Connell, S. T., and Thompson, P. A., 1995. "Molecular dynamics–continuum hybrid computations: A tool for studying complex fluid flows". *Physical Review E,* **52**, pp. R5792–R5795.

[11] Delgado-Buscalioni, R., and Coveney, P. V., 2004. "Hybrid molecular-continuum fluid dynamics". *Philosophical Transactions of the Royal Society London A,* **362**(1821), pp. 1639–1654.

[12] Wagner, G., and Flekkøy, E. G., 2004. "Hybrid computations with flux exchange". *Philosophical Transactions of the Royal Society London A,* **362**(1821), pp. 1655–1665.

[13] Nie, X. B., Chen, S. Y., E, W., and Robbins, M. O., 2004. "A continuum and molecular dynamics hybrid method for micro- and nano-fluid flow". *Journal of Fluid Mechanics,* **500**, pp. 55–64.

[14] Werder, T., Walther, J. H., and Koumoutsakos, P., 2005. "Hybrid atomistic-continuum method for the simulation of dense fluid flows". *Journal of Computational Physics,* **205**(1), pp. 373–390.

[15] Macpherson, G. B., Borg, M. K., and Reese, J. M., 2007. "Parallel generation of molecular dynamics initial configurations in arbitrary geometries". *Molecular Simulation,* **33**(15), pp. 1199–1212.

[16] Macpherson, G. B., and Reese, J. M., 2008. "Molecular dynamics in arbitrary geometries: parallel evaluation of pair forces". *Accepted by Molecular Simulation*.

[17] Macpherson, G. B., Nordin, N., and Weller, H. G., 2007. "Particle tracking in unstructured, arbitrary polyhedral meshes for use in CFD and molecular dynamics". *Under review for Communications in Numerical Methods in Engineering*.

[18] Smith, W., 1991. "Molecular dynamics on hypercube parallel computers". *Computer Physics Communications,* **62**(2–3), pp. 229–248.

[19] Rapaport, D., 1991. "Multi-million particle molecular dynamics. II. Design considerations for distributed processing". *Computer Physics Communications,* **62**(2–3), pp. 217–228.

[20] OpenFOAM: The Open Source CFD Toolbox. http://www.openfoam.org.

[21] Hertzog, D., Ivorra, B., Mohammadi, B., Bakajin, O., and Santiago, J., 2006. "Optimization of a microfluidic mixer for studying protein folding kinetics". *Analytical Chemistry,* **78**(13), pp. 4299–4306.

[22] Karypis, G., and Kumar, V. METIS. A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Version 4.0. University of Minnesota, http://glaros.dtc.umn.edu/gkhome/views/metis.

[23] ParaView: Parallel Visualization Application. http://www.paraview.org.

[24] Johnson, J. K., Zollweg, J. A., and Gubbins, K. E., 1993. "The Lennard-Jones equation of state revisited". *Molecular Physics,* **78**(3), pp. 591–618.