Parallel DSMC Application For The Analysis Of Argon Gas Flow In a T-Shaped Micro Manifold

Nevsan Sengil and Fırat O.Edis

Faculty of Aeronautics and Astronautics, Istanbul Technical University
Maslak 34469 Istanbul, Turkiye

Abstract. Direct Simulation Monte Carlo (DSMC) method uses representative molecules to simulate real gas flows. The computation time for the DSMC method mostly depends on the number of the representative molecules and the cells used in the computational domain. To realize the micro gas flow calculations in a shorter period of time, it is appropriate to divide the computational domain into multiple zones and analyze every one of them on separate processors in parallel. Consequently, molecule transfers are needed among neighbor zones. In our study we use Single Program Multiple Data (SPMD) programming model and blocking MPI routines among gas flow zones. Our results were compared with other studies from literature.

Keywords: DSMC, micro gas flow, SPMD, MPI
PACS: 51.10.+y

INTRODUCTION

In the last 25 years a number of Micro Electro Mechanical System (MEMS) were developed. These MEMS devices not only include the mechanical systems but also the fluids. Knowledge about fluid flows in this scale is not as mature as the mechanical properties of the MEMS[1]. As their dimensions are between 1 mm and 1 micron [2], gas flows related with the MEMS devices have higher Knudsen numbers (Kn) similar to high atmosphere flights. If Kn is higher than 0.1, instead of the classical continuum based Euler or Navier-Stokes (N-S) equations, deterministic or stochastic atomistic models should be used. This is due to the departure from local thermodynamic equilibrium with increasing Kn number[3]. Consequently, both the linear relation between shear stress and velocity gradient, and linear relation between heat conduction and temperature gradient are lost. DSMC method is a stochastic atomistic simulation method that can be used for high Kn number gas simulations. Our application of DSMC starts with the division of the computational domain into smaller cells. Linear dimensions of these cells are of the same order as the mean-free-path (λ) of the gas. A group of physical gas molecules are represented by one representative molecule. Every representative molecule carries position, velocity, cell number and if applicable, rotational energy information on it. Molecule movements and collisions are separated from each other. As a first step, molecules move according to their velocities and initial conditions. Their velocities, positions and cell numbers are updated after each movement step. In the collision step, stochastic approach is used and molecule velocities are updated according to the collision model chosen. Next step is the calculation of the macroscopic gas flow properties for each cell from the microscopic molecule information. For steady flows, time averaging is used for the calculation of macroscopic properties. In this study, a new, 2-D parallel implementation of DSMC method is developed. Later a micro-nozzle and T-shape micro manifold Argon gas flows has been analyzed for verification purposes. A gas flow through the micro nozzle is chosen because they are important for the micro propulsion systems for guiding the micro and nano satellites[4]. The computation time for the DSMC method strongly depends on the number of representative molecules and computational cells used in the flow area. To realize the gas flow calculations in a shorter period of time, it is necessary to divide the computational domain into multiple zones and analyze each one of them on separate processors in parallel. The final part includes results obtained with the new solver and comparisons with reference computations [5] along with the solution times of the parallel implementation.

ONE ZONE SEQUENTIAL DSMC SOLVER

For gas molecule collisions and sampling purposes gas flow area is divided to computational cells as shown in Figure 1 similar to the CFD meshes. The linear dimensions of the cell's are at the same order with the local mean-free-path (λ) of the gas flow[3]. A ratio of $(\lambda/3)$ has been taken as cell linear dimension. In DSMC method to keep computation time low, representative molecules[7] are employed instead of physical molecules. One representative molecule represents many real physical molecules. As number densities of the molecules can be change with position, the highest number density of the gas flow area taken into account for the cell linear dimension. By doing this it is aimed to keep at least 20 representative molecules in the cells. This number of molecules per cell will provide a reasonable statistical error for the calculation of macroscopic properties of the gas. Body fitted quadrilateral structural meshes are used to keep computational algorithms simple and fast. Although it was stated that this type of mesh is very problem specific[8], going to the multi-zone parallel implementation eases those concerns.

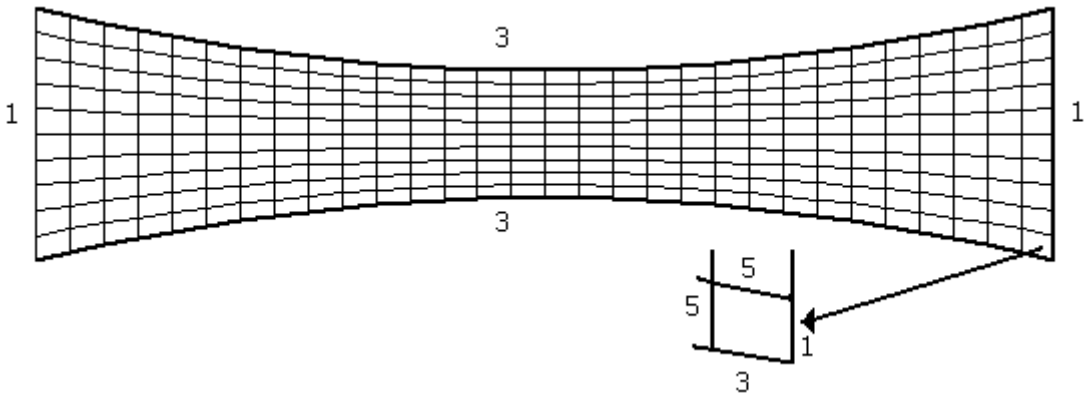


FIGURE 1. Body fitted quadrilateral structural meshes and boundary types in the convergent-divergent nozzle.

Zone boundaries are marked with unique integers as 1 and 3. Wall condition is denoted by 3, gas inflow or outflow is specified by 1. Cell boundaries are marked like zone boundaries. In addition to the zone boundary types, cell boundaries have one more boundary type that is marked with the integer 5. This one is for internal boundary between cells.

Cell data is placed in a two-dimensional array. Those data are composed of macro values like pressure, temperature, density, flow velocity components and cell volume. Initially, representative molecules are placed into the cells. Their numbers and velocities are calculated from the cells' initially imposed densities and temperatures. For initial velocity components (u, v, w) Maxwell velocity distribution is used.

$$\begin{aligned} u &= \sqrt{-\ln(r)\,2kT/m}\; Sin(2\pi r) \\ v &= \sqrt{-\ln(r)\,2kT/m}\; Sin(2\pi r) \\ w &= \sqrt{-\ln(r)\,2kT/m}\; Sin(2\pi r) \end{aligned} \tag{1}$$

Here k, T, m are Boltzmann constant, cell temperature, and molecule mass by order. r is a random number between 0 and 1. Molecules' initial positions are chosen as random in the cell. First, a molecule is positioned in a rectangle that its corner coordinates are the maximum and minimum x and y values of the quadrilateral cell. Then molecules are tested for they are inside or not in the cell. If not, a new molecule is positioned randomly and tested again. Until the determined number of representative molecules are place in the cells procedure is kept continuing. This procedure is very similar to the acceptance-rejection technique.

Another array contains molecule data such as molecule velocity components, molecule positions, and rotational energy if applicable. This data is organized to point to the cell number, and the unique molecule number within this cell. Once representative molecules are initially placed into the gas flow zone, DSMC solver moves the

representative molecules to their new positions using a time step and given molecule velocity. Time step is determined such that a molecule cannot cross many cells within each time step[7]. In this step representative molecules can stay in the same cell, can move to a neighboring cell, can be reflected from the walls or leave the zone. If it is determined that the molecule is not in the same cell anymore, the edge from which the molecule left the cell is determined. Depending on the boundary type of the edge, molecule can be traveling to a neighbor cell, reflected from a wall or leave the zone. Molecules that left the zone are deleted. Other molecules are traced until the end of their journey. Once they reach their final cell they are placed a new array called NEWMOLLS(M,N,P). At the same time new representative molecules are introduced into the zone from the inflow or outflow boundaries assuming the equilibrium conditions. Those molecules are also placed into the NEWMOLLS(M,N,P) array. At the end of the molecule movement phase, NEWMOLLS(M,N,P) is copied onto the MOLLS(M,N,P). As a result no cross-reference array linking the molecules to the cells is required. This algorithm is similar to the cell-based methods explained in the references[6, 9].

Next step is the collision of the molecules that are located in the same cell with each other according to the gas kinetic principles. As a result of binary collisions, the velocities of the molecules are updated. Total momentum and total energies are conserved. The positions of the molecules are not changed. In this DSMC solver VHS(Variable Hard Sphere) collision model and NTC(No Time Counter) scheme[3] is used. For rotational and translational energy exchanges for the diatomic molecules, Borgnakke-Larsen model is added.

Finally, macro values are obtained from the molecule velocity and number density information. These data are also used for completing and updating the boundary conditions at every predetermined time period.

PARALLEL IMPLEMENTATION

Because of the high computational cost of the DSMC method[6], certain measures are needed to shorten the computation time. One way of doing this is to use a domain decomposition method. Gas flow zone is divided to a number of sub zones. Here every sub zone is appointed to a processor and numbered with an integer greater than 10. If a sub zone has a neighbor sub zone then the mutual boundary is marked with the number of this neighbor sub zone. In parallel implementation the molecules that cross the zone boundary are stored in an array. Every processor performs the calculations for one time step and waits the others for synchronization. Representative molecules leaving the sub zones are transferred to the new processor. Molecule number densities and cell dimensions can be different in the sub zones, so is the ratio of the physical to representative molecules.

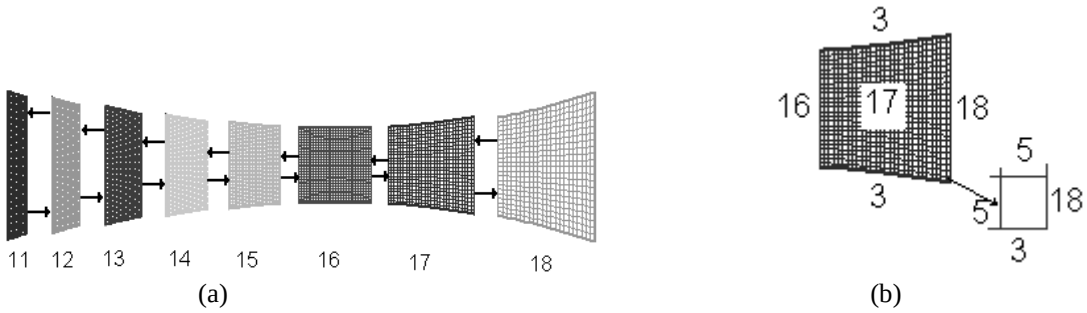


FIGURE 2. a) Multi zone convergent-divergent nozzle for parallel DSMC application. b) Sub zone boundary marking

MPI communication scheme among processors is preferred for better portability. Blocking data transfer routines are employed for simplicity and easy code debugging. Although these routines are easy to implement some care is needed. Here all sub zones are marked so that the neighbor of the odd number ones are even numbers or vice versa. Once the transfer arrays are obtained, first odd number sub zones send the output data and wait for the input data. Second, even number sub zones receive the output data. Following that they send their output to the odd number sub zones.

Once the number of available processors is determined, a cell mesh data file and a boundary condition data file are prepared for every processor. Same DSMC code is run on all processors. This structure can be considered as an SPMD (Single Program Multiple Data) programming scheme.

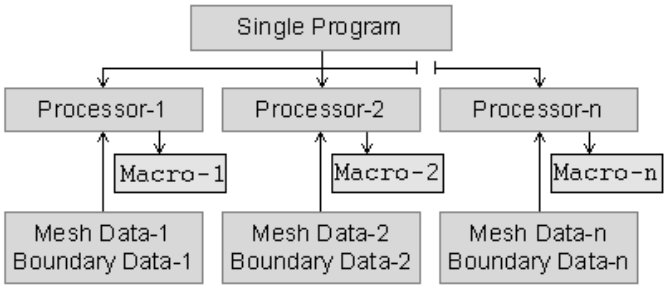
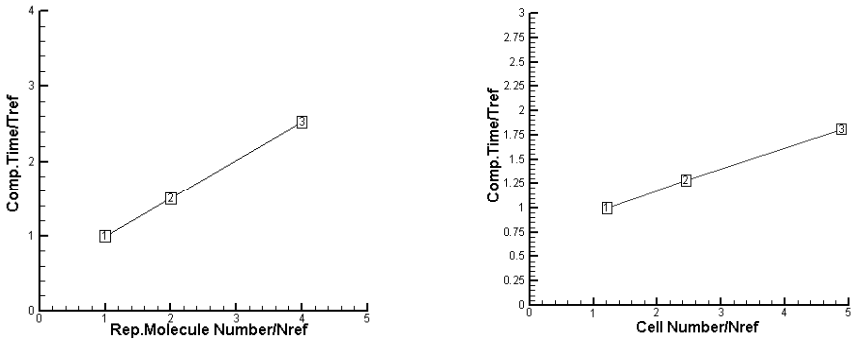


FIGURE 3. SPMD (Single Program Multiple Data) data structure used in this DSMC solver

As stated before when the molecules are transferred to the neighbor zone, there is a possibility that some of them are reflected back or by-passed the zone they sent and reach a different zone. So a control loop is employed for the inter-zone communication. Each zone inform the others when molecule movements are completed. Only then the inter-zone communication is ended. Finally, every processor outputs a data file that includes the macro data of the sub zone analyzed on that processor.

In parallel programming, load balancing among processors is crucial. For maximum efficiency total load should be divided as even as possible. Computational load for each sub zone mostly depends on the number of representative molecules and the number of cells in that region[9]. In this study, static load balancing is applied with the help of a preprocessor. This requires an iteration process in the preprocessor until the load is shared equally. To estimate the load coefficients of the representative molecules and the number of cells a number of test cases were run. Feeding these load coefficients and the boundary conditions to the preprocessor, a required number of sub zone mesh data and boundary conditions corresponding to optimal load distribution are obtained. Because of the varying molecule number densities, every sub zone can have different ratio of physical to representative molecules, cell numbers and linear cell dimensions. As a result of this, total computational load of multi zones can be lower than the total computational load of one zone for the exactly same gas flow area. Consequently, DSMC method is declared a promising candidate for parallel computing[7].



(a) (b)

FIGURE 4. Variation of computational time with: a) Representative molecule numbers. b) Cell numbers.

When a molecule crosses to a sub zone with different ratios of physical and representative molecules, some extra measures should be taken. First, some molecules should be deleted or cloned[6]. Second, deleting should be done before the communication while copying should be done after the communication to obtain a better efficiency figure.

VERIFICATION TEST CASES

Two test cases namely, the Argon flow in a convergent-divergent nozzle and flow in a T-shaped micro manifold are chosen from the literature and our results are compared with the reference[5] for verification purposes. As shown in Figure 5 and Figure 6, results agree very well with the reference. Solution times of the parallel implementation for the nozzle test case are demonstrated in Table 1. Although the total number of the representative molecules is kept around 990.000 in all instances, the cell numbers are different.

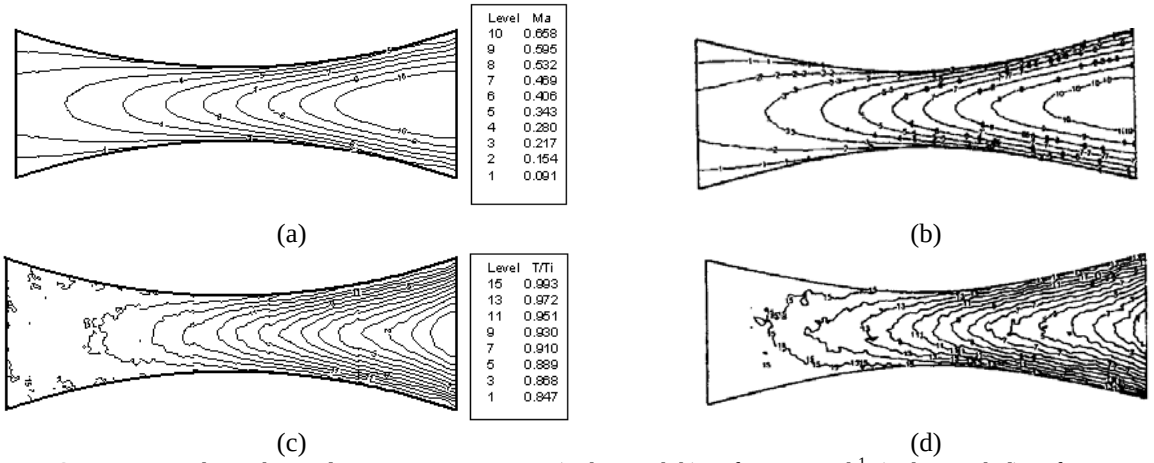


FIGURE 5. Mach number and temperature contours a) This work b) Reference work[1] c) This work d) Reference work[1]

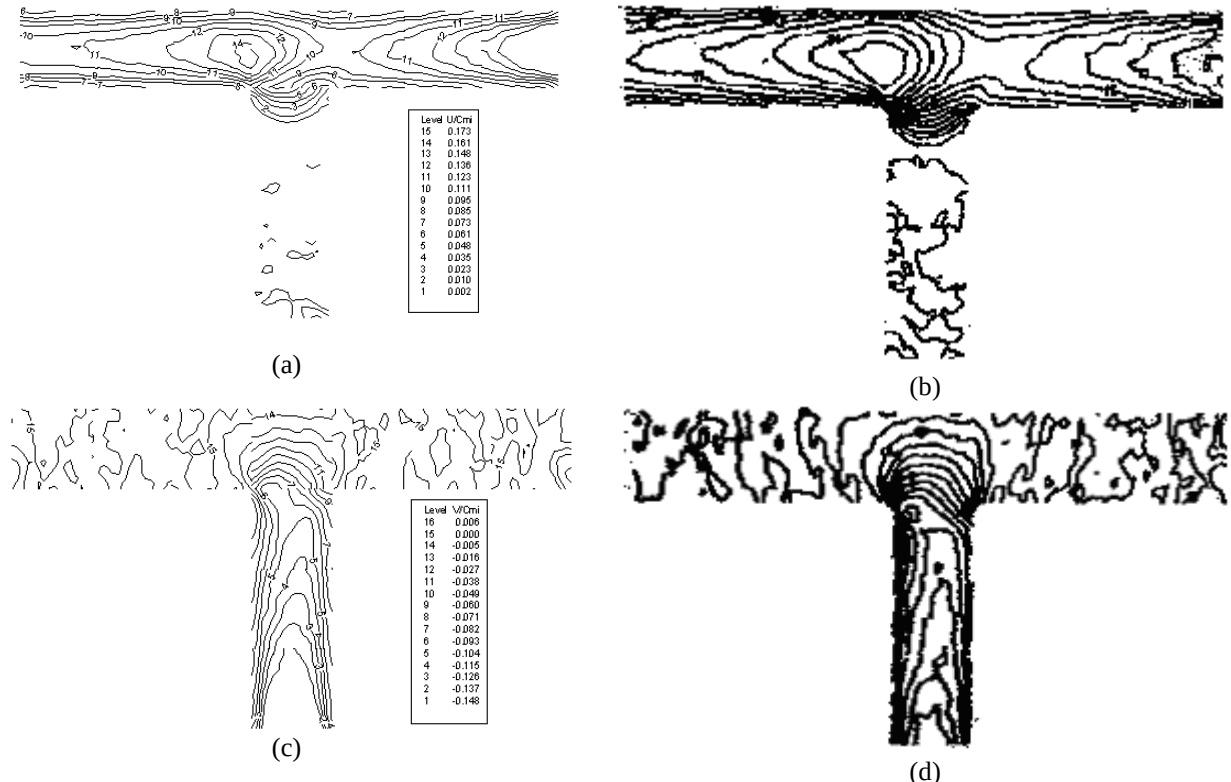


FIGURE 6. Velocity contours (a) This work horizontal velocity contours (b) Reference work[1] horizontal velocity contours (c) This work vertical velocity contours (d) Reference work[1] vertical velocity contours

[1] Reprinted from Computers &Fluids 30(2001),J.-S. Wu, K.-C. Tseng, Analysis of micro-scale gas flows with pressure boundaries using direct simulation Monte Carlo method, © 2001, with permission from Elsevier

TABLE 1. Convergent-divergent Nozzle Solution Times for Multi-processor Application on SGI Altix 3000 Series Computer with 32 Itanium II (1.3 GHz) processors and 64 GB Memory

Number of Processors	**1**	**2**	**4**	**8**
Solution Time (sec)	63.666	26.103	12.879	7.096
Representative Molecule Number	~990.000	~990.000	~990.000	~990.000
Cell Number	24.843	13.580	10.111	7.835

CONCLUSIONS

First a new 2-D sequential DSMC solver has been developed using the gas kinetic principles to analyze micro-gas flows. Different number of the representative molecules and cells are employed to determine their effects on the computing time and processor loads. This data is used to develop a preprocessor code for sharing the load as evenly as possible among the processors for parallel applications. Static load distribution is employed in our 2-D parallel DSMC solver. Two test cases from literature are selected and our DSMC solver is run for verification purposes. Similar results are obtained. Parallel runs using up to 10 processors were performed. Parallel solution times were obtained for the 2, 4 and 8 processors for the nozzle flow test case.

It is concluded that for a better load sharing among the processors to employ dynamic load balancing could be more promising instead of static load balancing. When the load is shared statically, processor loads can change during the analysis. This stems from the dynamic character of the DSMC method. For a better load sharing, an active monitoring policy including a continuous meshing is needed[11]. In the future, moving boundary capability will be added to the solver and 3-D implementation will be performed.

REFERENCES

1. E.S. Piekos, K.S.B., *Numerical Modeling of Micromechanical Devices Using the Direct Simulation Monte Carlo Method.* Transactions of the ASME, 1996. **118**(September 1996): p. 464-469.
2. M.Gad-el-Hak, *Flow physics in MEMS.* Mecanique and Industries, 2001. **2**(2001): p. 313-341.
3. G.A.Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. The Oxford Engineering Science. 1994, Oxford: Clarendon Press.
4. Peng-Fei Hao, Y.-T.D., Zhao-Hui Yao, Feng He, Ke-Qin Zhu, *Size effect on gas flow in micro nozzles.* Journal of Micromechanics and Microengineering, 2005. **15**(2005): p. 2069-2073.
5. J.-S. Wu, K.-C.Tseng, *Analysis of Internal micro-scale gas flows with pressure boundaries using the DSMC method. Computers&Fluids,2001.* **30** *(2001): p. 711-735.*
6. Stefan Dietrich, I.D.B., *Scalar and Parallel Optimized Implementation of the Direct Simulation Monte Carlo Method.* Journal of Computational Physics, 1996. **126**(1996): p. 328-342.
7. LeBeau, G.J., *A parallel implementation of the direct simulation Monte Carlo method.* Comput. Methods Appl. Mech. Engnr., 1999. **174**(1999): p. 319-337.
8. J.-S. Wu, K.-C.T., T.-J.Yang, *Parallel Implementation of DSMC Using Unstructured Mesh.* International Journal of Computational Fluid Dynamics, 2003. **17**(5): p. 405-422.
9. Min Gyu Kim, H.S.K., Oh Joon Kwon, *A parallel cell-based DSMC method on unstructured adaptive meshes.* International Journal for Numerical Methods in Fluids, 2004. **44**(2004): p. 1317-1335.
10. Intel, *Intel Math Kernel Library Reference Manual*. 2004.
11. Robert P.Nance, R.G.W., Bongki Moon, H.A.Hassan, Joe Saltz. *Parallel DSMC Solution of Three-Dimensional Flow Over a Finite Flat Plate*. in *6th AIAA/ASME Joint ThermoPhysics and Heat Transfer Conference*. 1994. Colorado Springs, CO: AIAA.