Abstract

The plume of a thruster firing into a vacuum rapidly expands to extremely low densities. Under these rarefied conditions the number of inter molecular collions is insufficient to maintain local equilibrium in the flow. As a result, the assumption of continuum fluid mechanics become invalid and the gas cannot be properly modelled as a continuum fluid. In such a case a microscopic approach which considers the particle nature of a gas is appropriate. In our case we have employed the direct simulation Monte Carlo (DSMC) Method initially developed by Bird.

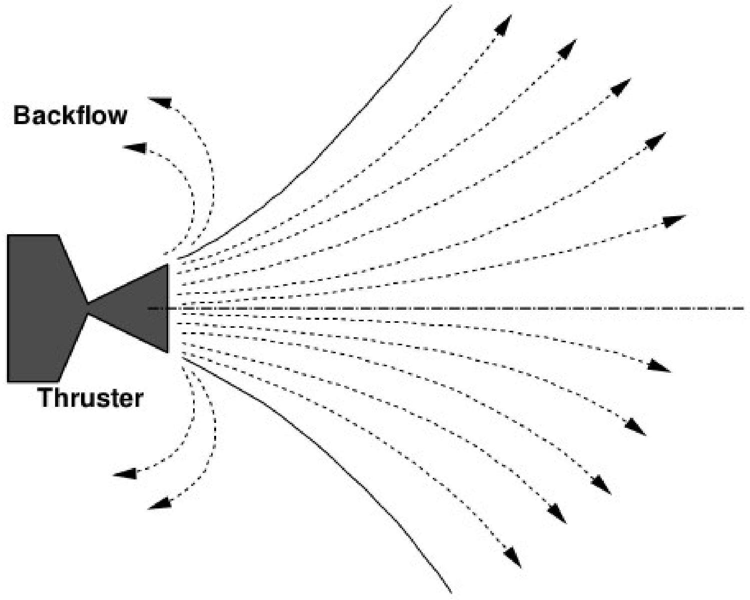
The goal of this work is to validate the capability of dsmcFoam in simulating the flow of nozzle plume for a relatively small rocket nozzle and subsequent comparison with the widely used Simon method . The results and learning from this would be used in the future to simulate nozzles of much bigger dimensions. DSMC method is computationally expensive and parallel computation is required for simulating nozzle exhaust. Thus it is desirable to develop the Simon Method so as to save us from running dsmc simulation for various exit flow conditions.

Table of Contents

1. Introduction

Spacecraft in orbit require propulsion systems for various functions including attitude control and station keeping. Attitude adjustments may be necessitated by such mission requirements as the pointing of antennas or scientific instruments. Power requirements may require attitude adjustments to ensure optimal sun coverage of solar arrays. Deceleration caused by orbital drag must be accounted for in order to maintain a stable orbit.
Low thrust rockets are often employed to meet these on orbit requirements. The impulse needed for these tasks is generally small so that small thrusters can handle the task. An array of thrusters placed around the spacecraft ring in different directions is needed to provide the full range of orbital and attitude control. Each task uses a number of thrusters ring in unison. The use of more than one thruster allows greater control of the forces and torques exerted on the spacecraft.

When a control thruster is fired, a plume of exhaust gases is produced. While the bulk flow is directed along the axis of the thruster, the plume will also expand radially. In a vacuum environment, this radial expansion can be very broad. There will also be a noticeable amount of back flow gas which flows behind the thruster.

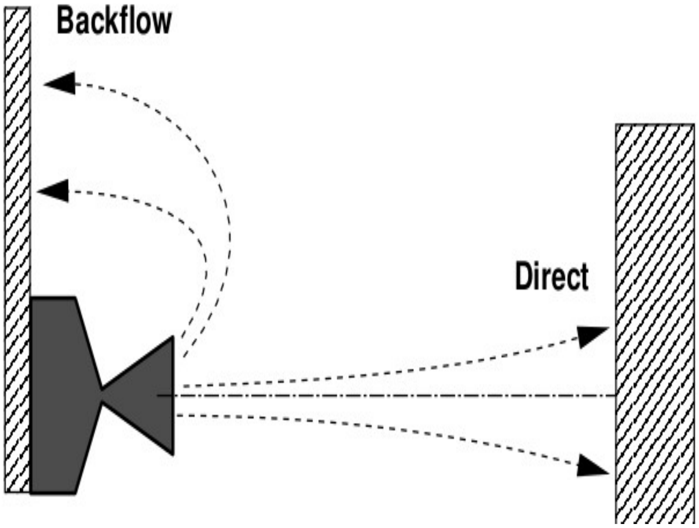


Thruster Plume

Impingement of a plume on a spacecraft can have a number of undesirable effects. Momentum transfer results in additional forces and torques which affect the attitude of the spacecraft. Depending on the orientation of the spacecraft, its component surfaces and the plume, these forces can be a significant fraction of the total thrust. The propulsion system must then compensate for these forces at the cost of additional propellant and possibly electrical power. Increasing the amount of propellant required means either the lifespan of the spacecraft is decreased or more propellant must be carried, reducing the amount of mass available for payload.

When a plume molecule strikes the spacecraft it transfers kinetic and internal energy to the surface. In most cases this results in heating of the surface. Thermal control is an important part of spacecraft design. Many electronic components can only function properly in a range of temperatures. The thermal loads caused by plume impingement must be accounted for and dissipated by the thermal control system.

The flux of particles striking a surface can have detrimental effects on the surface. Many of the gas species which are used as propellants can be classified as contaminants.



Impingement interaction with spacecraft surface

2. Direct Simulation Monte Carlo

The Direct Simulation Monte Carlo is a particle based method for the simulation of nonequilibrium gas flows. The gas is modelled at the microscopic level using particles which of each represent a large number of real atoms. The physics of the gas is modelled through motion of particles and collisional interactions between them. The method is statistical by nature.

A microscopic approach to the simulation of gas flows can be advantageous due to the ability to capture non equilibrium effects. The use of continuum fluid mechanics is based on the assumption that the gas is in local equilibrium. That is, the velocities and energies of the gas particles are distributed according to equilibrium distribution functions. This allows the gas to be modeled as a continuum fluid using such thermodynamic quantities as temperature. In order to maintain equilibrium there must be sufficient collisions occurring in the gas. This requires that there on an average the number of particles in an differential volume be constant.In many flows of interest this is not the case and the gas cannot be modeled as a continuum fluid without neglecting non nequilibrium effects.

By considering the gas on the microscopic level as a collection of discrete particles the DSMC method is able to capture non equilibrium effects directly. The non equilibrium nature of a problem can be measured using the Knudsen number. This non dimensional parameter is the ratio of the mean free path length in the gas to the length scales of the flow:

$Kn = \lambda/L$ (In our case L would be the diameter)

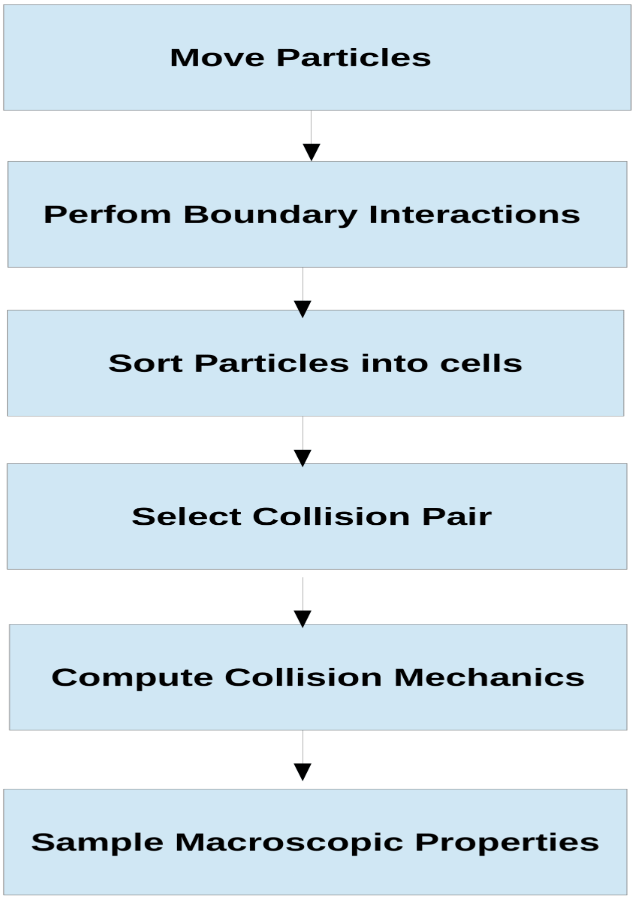
The mean free path is the average distance travelled by a gas particle between collisions. If the length scales of the flow are comparable to the mean free path then there will be insufficient collisions to maintain equilibrium. Flows with Knudsen numbers on the order of 0.01 or smaller are continuum flows and may be modeled using more conventional fluid mechanics.

A Knudsen number greater than 3 indicates the flow is essentially free molecular, where the number of intermolecular collisions is small enough to be neglected. Between these extremes are transitional flows where collisional behavior is important but there are insufficient collisions to ensure local equilibrium. Examples of such transitional flows include the rapidly expanding gas plume from a thruster ring into a vacuum or the shock interaction in front of a hypersonic vehicle.

2.1 Algorithm

The DSMC technique models a gas by following the trajectories of gas molecules and computes the effects of collisions between particles. One of the fundamental assumptions of the DSMC method is that particle movement can be decoupled from collisional behavior. Particles then move along straight line paths with discrete changes in velocity and energy caused by collisions. In order to accomplish this, the simulation is broken up into time steps the magnitude of which are small compared to the average time between collisions. In this small interval of time, these two processes can realistically be treated as independent.

A complete DSMC simulation is a series of discrete time steps or iterations. During each iteration particles are moved and collisions are computed. The algorithm for a single iteration can be broken down into the following operations:


Move Particles
Perfom Boundary Interactions
Sort Particles into cells
Select Collision Pair
Compute Collision Mechanics
Sample Macroscopic Properties

2.2 Molecular Motion

Particles are moved through space during each iteration according to their individual velocities and the time step. In the absence of external force fields such as gravitational or electromagnetic fields particles follow straight line trajectories. The change in particle position is then given by:

$$\Delta x = v^* \Delta t$$

This simple vector equation is sufficient for simulations in one, two or three dimensions. Although particles have velocity components in all three directions regardless of the dimensionality of the problem, motion in the homogeneous direction or directions is ignored in two- and one-dimensional simulations.

2.3 Molecular Dimensions

The standard number density at a pressure of 1 atm a temperature of 273K is 2.6 e25. It can be calculated from the following formula .

$$PN_A = nRT$$

The average molecular diameter for air as calculated from the VHS is 4.5 e -10 m. The mean free path for air molecules is 4.9 e -8 m. It can be calculated from the follwing formula

$$\lambda = \frac{1}{\sqrt[2]{2}\pi d^2 n}$$

The average diameter is inversely proportional to the one-eighth power of temperature . Thus, mean free path is inversely proportional to the number density and also inversely proportional to the one – fourth power of temperature . The viscosity coeeficient is proportional to temperature raised to the power three – fourth. The mean collision rate comes out to be 7.3 e9.

2.4 Boundary Interactions

2.4.1 Symmetric walls

This corresponds to usage of specular reflection. Specular reflection is mirror like reflection . It happens when

- The metal surface has been outgassed through exposure to high vaccum and temperature.
- When the ratio of molecular weights of the gas to that of the surface molecules is small in comparison to unity.

When a particle hits this boundary, the components of velocity normal to the wall is reversed and the tangential component is kept the same.

2.4.2 Diffusive walls

In diffusive walls the velocity of the reflected molecule is independent of the velocity of incident particle. A particle striking a fully diffuse surface is accommodated to the surface temperature. Outgoing velocity components sampled from Maxwellian distribution at that temperature. No memory is retained of the incoming properties and the outgoing direction is random. Real surfaces fall somewhere between these two extremes. To simulate this, a surface can be considered partially accommodating, with a fraction of incoming particles refecting specularly while the rest are diffusely refected. A parameter is defined to represent the fraction of diffuse reflections to specular reflection is the tangential momentum accomodation coefficient.

2.4.3 Periodic walls

A molecule that crosses a periodic wall is send back into the domain through the corresponding point on the other periodic wall.

2.4.4 Outflow boundary walls

A molecule that crosses an outflow boundary wall is discarded from the molecule list

2.4.5 Inflow boundary wall

-
The flux of particle at inlet is computed considering Maxwell's distribution outside the boundary edge. During every time step, the molecules that have a component of velocity normal to and into the flow domain are inserted into the domain.

2.5 Computational parameters

2.5.1 Particle weight

Each particle in a DSMC simulation represents a large number of real molecules or atoms. The ratio between the number of real particles and simulation particles is termed the particle weight.

$$W_p = N_r/N_s$$

Smaller values of this ratio result in a larger number of computational particles and thus better resolution of flow physics. Increasing the particle weight decreases the number of particles and thus decreases the cost of the simulation. The value of the particle weight is selected so as to obtain a balance between resolution and efficiency. Typical value for a two-dimensional simulation might range from 1 e11 to 1 e13. We shall sometimes refer it to as Nsim .

2.5.2 Time Step

A finite time step is used to decouple the movement of particles from collisions. In order to do this, the time step must be a small fraction of the mean time between collisions. The exact expression is dependent on the collision model employed but is in general a function of density and temperature. The mean collision time can be computed by dividing the mean free path by the mean magnitude of relative velocity of colliding molecules which is √2 times the thermal velocity.

The mean free path for a VHS model is

$$\frac{(5-2\omega)(7-2\omega)}{15}\sqrt{\frac{m}{2\pi kT}}\frac{\mu}{\rho}$$

and the mean thermal speed is

$$\sqrt[2]{\frac{8kT}{\pi m}}$$

2.5.3 Grid Cell Size

A computational grid is used in DSMC to group particles for the purposes of calculating collisions and sampling macroscopic properties. The size of the computational cells is limited by the constraints of physical accuracy and computational efficiency.

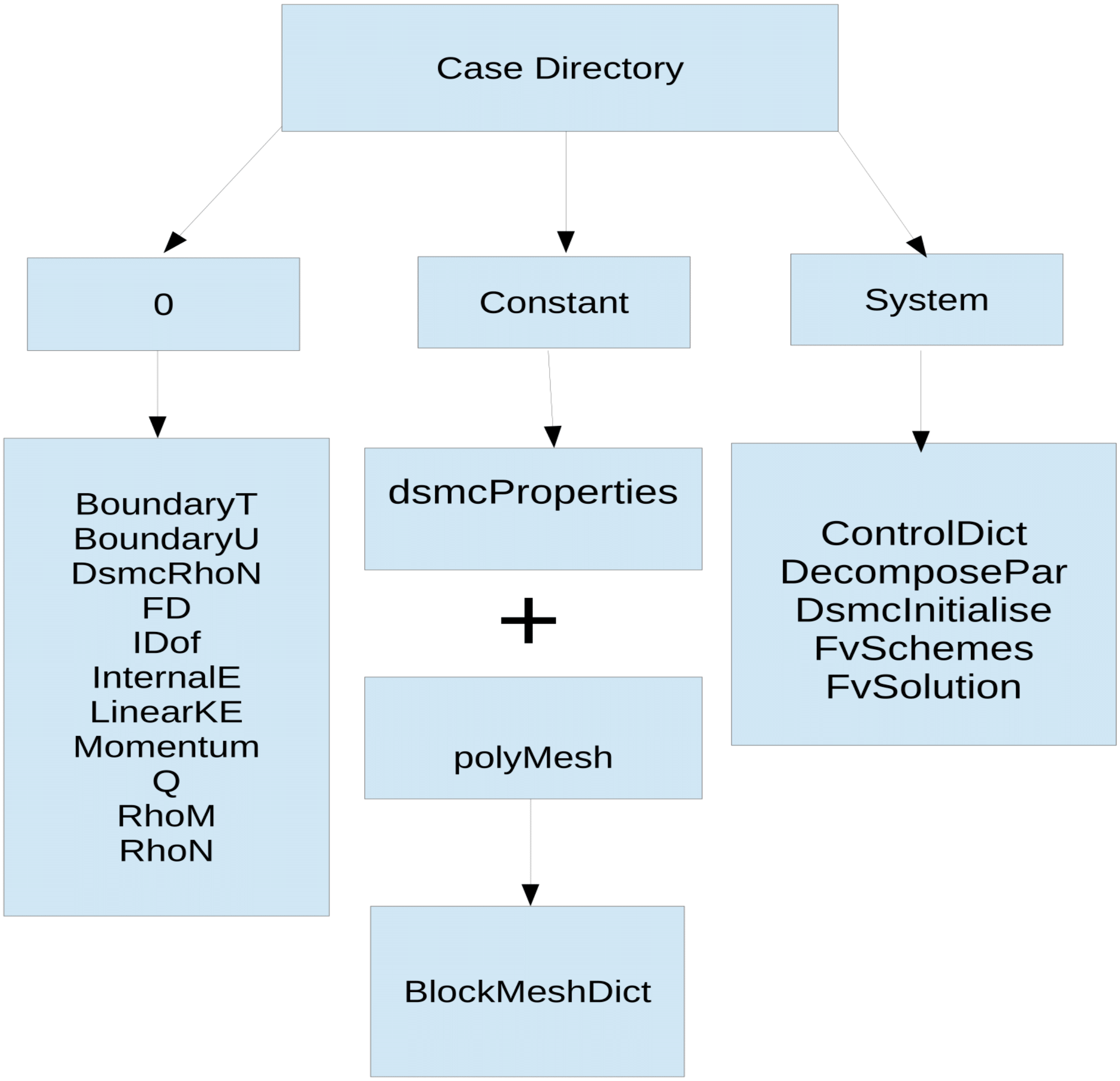
As previously mentioned, collisions are calculated statistically between pairs of particles residing in the same cell. The two particles participating in a collision may be separated in space by as much as one cell dimension. In a real gas the average distance travelled by a particle between collisions is termed the mean free path .

In a real gas two particles will not collide if they are separated by more than one mean free path. This provides a limit on the size of the computational cell. Ideally, cell dimensions should be smaller than the local mean free path (e.g. x < 0. 3* mean_free_path) at all points. The presence of cells which are larger than one mean free path results in reduced resolution of gradients at the macroscopic level.

Computational efficiency provides a lower bound on the size of cells. In order to accurately resolve collisional effects it is necessary to have a minimum number of particles in each cell. The smaller the cells that are used the larger the number required to fill the domain and consequently the larger the number of particles needed for the simulation. Hence an optimum number needs to be chosen for simulation to be viable.

3. OpenFOAM

The present project is based on the use and application of dsmcFoam. DsmcFoam is a direct simulation Monte Carlo simulation tool available in OpenFOAM. OpenFOAM is an open-source C++ library/tool developed by Jasak et al and maintained by OpenCFD Ltd. The current version is 4.0 . This tutorial describes the study of basic handeling of OpenFOAM, especially dsmcFoam and its use to simulate the exhaust plume of a small nozzle.



3.1 DsmcFoam

We shall describe in detail the process of setup, simulation and post -processing for dsmcFoam with an aim of orienting a new user to basic procedures of running dsmcFoam. OpenFOAM provides four basic tutorials for dsmcFoam which helps in understanding the solver better. The first step is to copy the contents of the tutorials to an empty directory so that it can be used multiple times. The tutorials are present in the OpenFoam directory and should be copied to a new directory within your Open foam installtion directory.

3.2 Nozzle Flow simulation

This tutorial will describe how to pre-process, run and post-process a case involving supersonic exhaust from a small nozzle(diameter 40 mm). Supersonic exhaust flow of air at a exit velocity of 1800 m/s is analyzed. VHS model was employed for particle collision. In diatomic Nitrogen molecular collisions are inelastic which results in energy exchange between the translational and rotational energy compoenents. The Larsen – Borgnakke energy model has been activated to model the energy interaction . The temperature of the exhaust is 860 K while that of the wall has been taken as 1000 K . The mean free path in VHS model is computed using the following formula

$$\frac{(5-2\omega)(7-2\omega)}{15}\sqrt{\frac{m}{2\pi kT}}\frac{\mu}{\rho}$$

where w is the temperature coefficient of viscosity , m is the atomic mass, k is the Boltzman constant, T is the temperature , meu is the gas dynamic viscosity and rho is the gas density.

3.3 Initial and Boundary Conditions

The values of boundary conditions are given in the 0 folder of the case directory.

Out	300 K	(1500 0 0) m/s
Wall	1000K	(0 0 0) m/s
sym	symmetryPlane	symmetryPlane
In	870 K	(1500 0 0) m/s
Minusz	empty	empty
Plusz	empty	empty
Front	300 K	(-1500 0 0) m/s
Top	300 K	(0 1500 0) m/s
Upstream	300 K	(0 -1500 0) m/s

Notice the value of the boundary velocity on all the patches. It has been deliberately selected in such a way such that all the particles created at patches except **In** are blown away from our domain. In all the other files(dsmcRhoN, rhoN, etc) the value of zero gradient is given.

3.4 Solver Controls

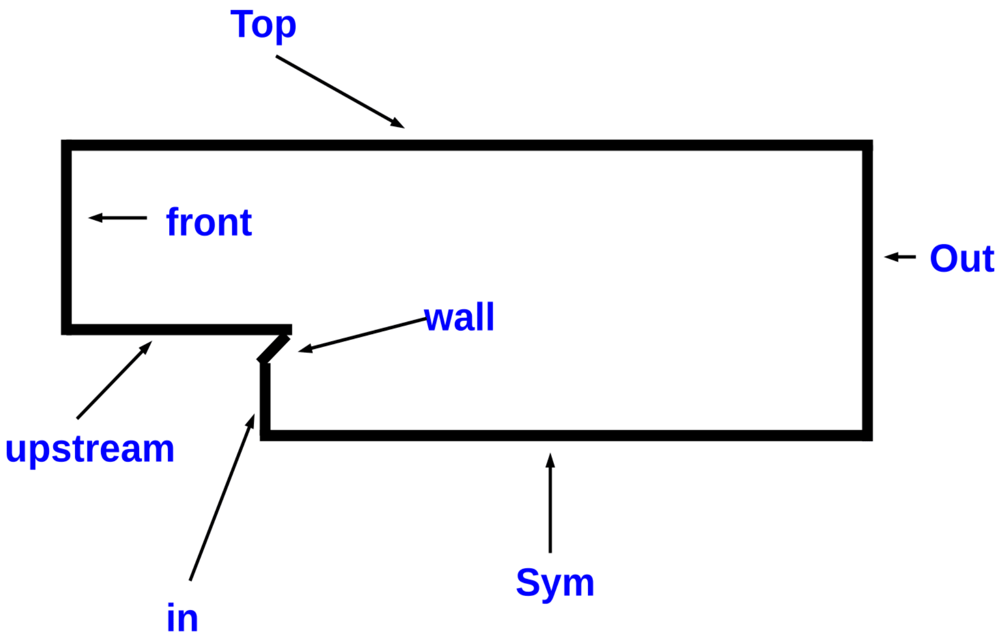
In the file system/controlDict the startTime is set to 0 and endTime is set to 0.01 since the solution is a transient one. Also writeInterval is set to 0.0001. The Δt given is 2e-8.

3.5 Particle Initialisation

For the purpose of particle initialisation the Number density taken is 1e16.

3.6 Geometry

The nozzle under consideration has an exit diameter of 40 mm. OpenFOAM takes all grids in 3D. We can however solve a 2D case by giving empty or cyclic boundary conditions. The grid under consideration has cells till 5 m downstream and 5m in the y direction. The grid has been stretched behind the nozzle to visualise any backflow. The z direction has a dimension of 1 mm. It has just one cell. Although we are doing a 2D case the volume in the z direction matters as we will discuss later. The mesh was created using pointwise software and exported to OpenFOAM format. The cells are non uniform and structured.



4. Results

4.1 Growth of flow

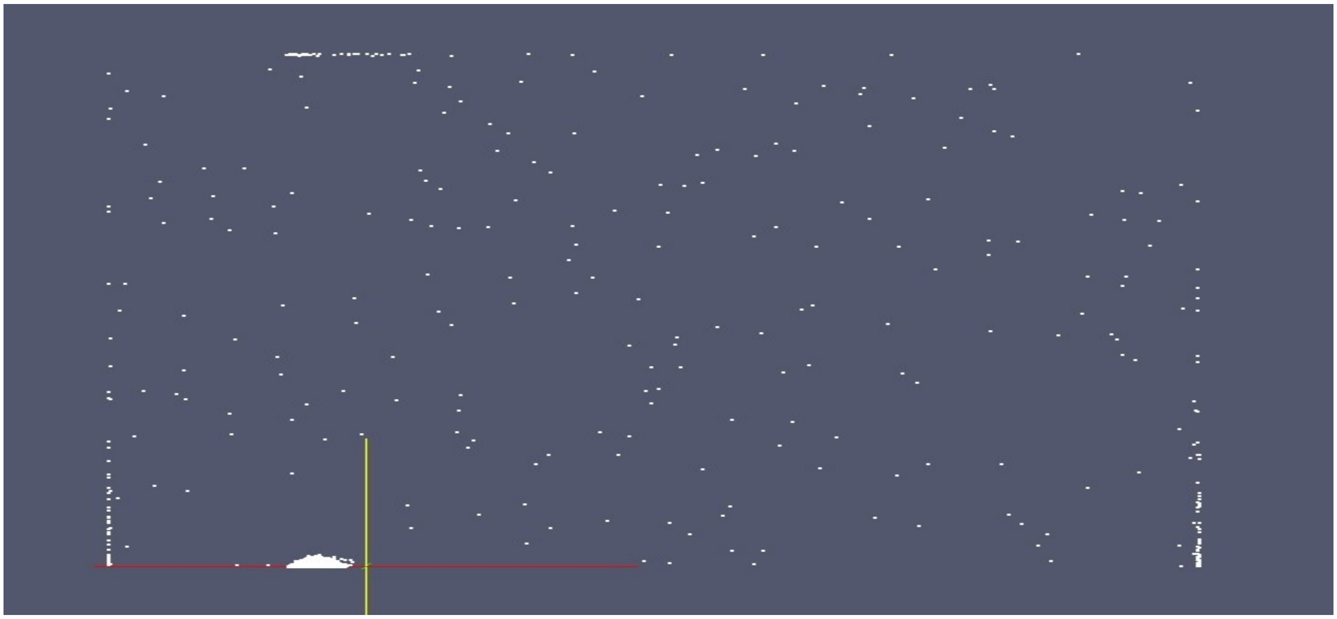


Illustration 1: Particles distribution at 1 millisecond

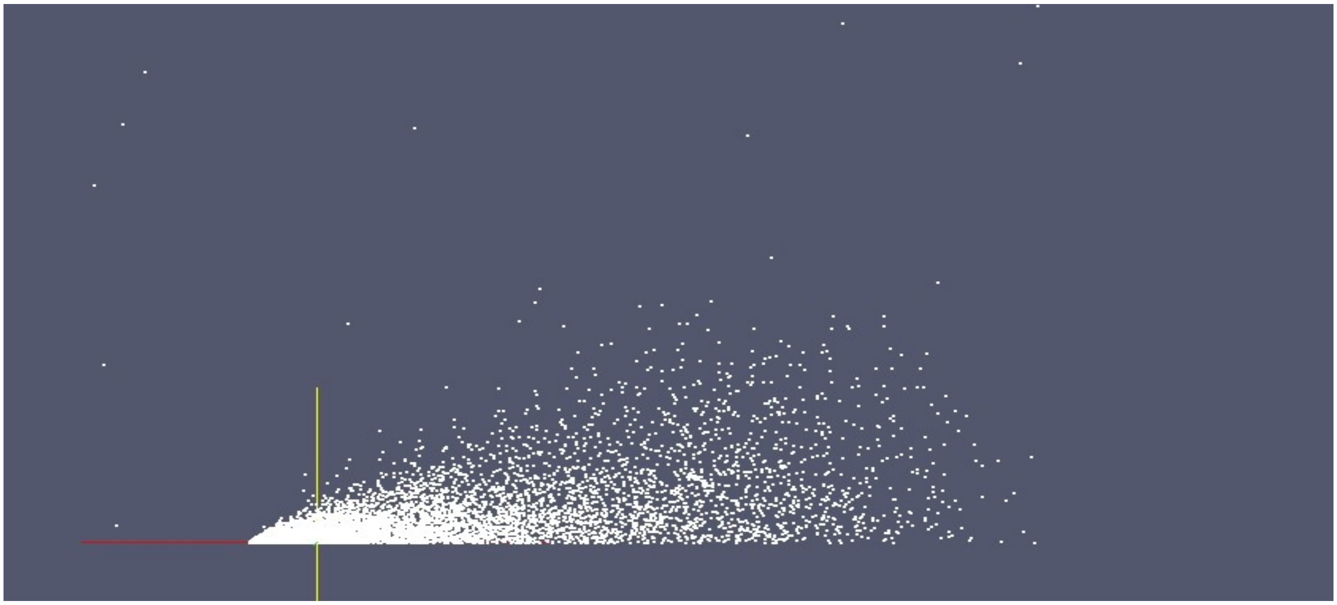


Illustration 2: Particle distribution at 2 millisecond

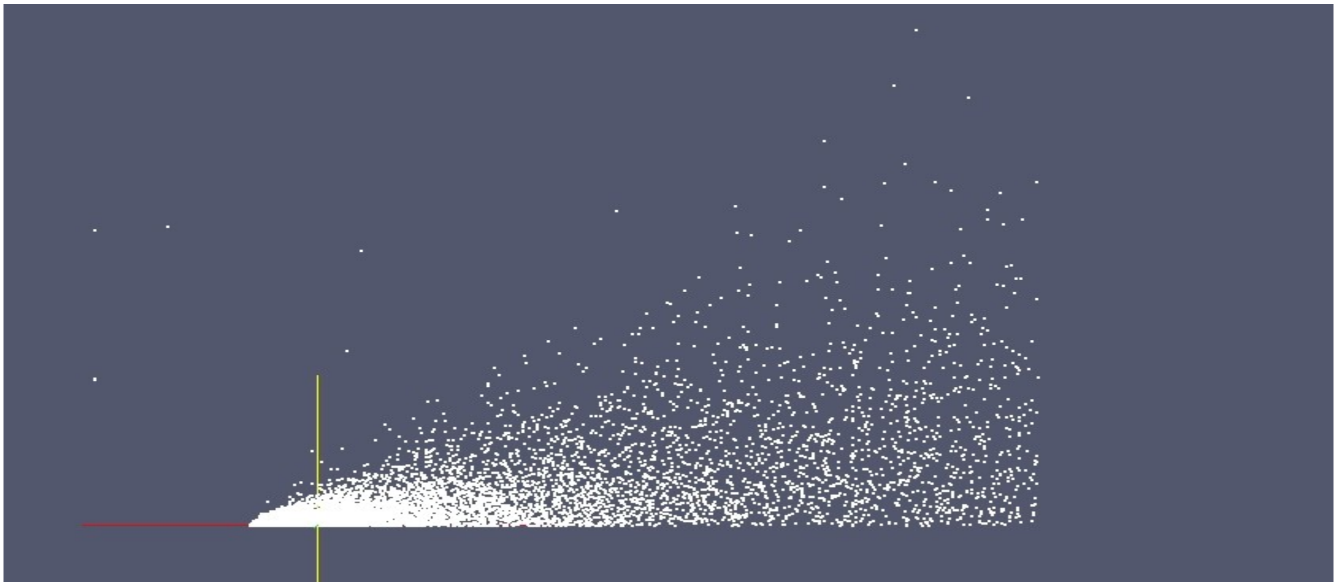


Illustration 3: Particle Distribution at 8.8 millisecond

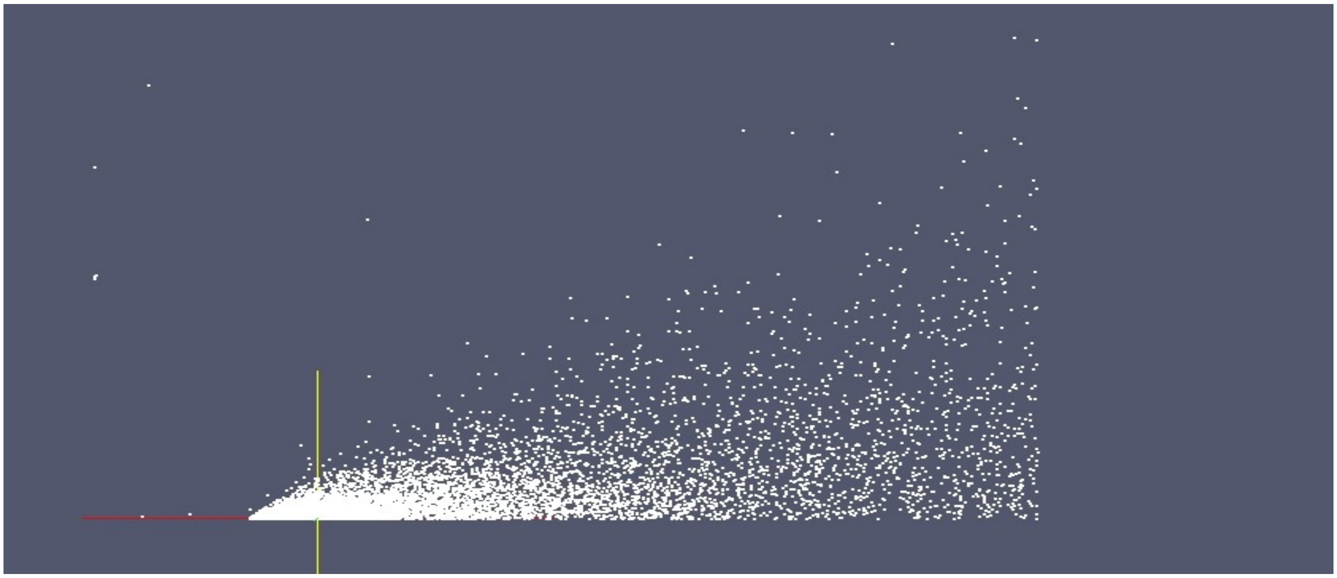


Illustration 4: Particle Distribution at 12 millisecond

5. Discussion

- The first simulation would give us an idea of the area where the maximum gradient change is occuring. Based on this the grid must be made finer in near exit and in the nozzle axis of symmetry and courser in areas away from the nozzle.

- The use of Unstructured grid seems to be a plausible solution to some of the problems related to number of particle per cell. In our case the cell size must increase in proportion to the mean free path to keep the number of particles to the requisite value.

- It should be noted that the required condition of 8 particles per cell is violated for cells lying more than 30 mm downstream from the nozzle exit. This happened because the cell sizes were close to 1mm . This points out to a very important part of dsmc which is grid manipulation. The first few simulation give us an idea of the kind of grid that is required for accurate results. Here the mean free path of the gas is much greater than 1mm and thus the number of particles are falling greatly. Since all the macroscopic properties of flow field are obtained by sampling all the particles within a cell, the number of particles per cell must be sufficient to minimize the statistical error. Clearly, the larger the particle number per cell ,the less the statistical error. It is quite evident that as the particle count reduces downstream for simulation with higher particle weight greater anamolies are seen in the values.

nEquivalentParticles is the number of real particles each simulated dsmc particle is representing. Good dsmc practice recommends it to witin a range of 1e9 to 1e15.

Wall interaction model has been choosen to be MaxwellianThermal. Other options include SpecularReflection and MixedDiffuseSpecular. When a particle strikes a specular surface, its componentts of velocity normal to the surface is reversed. This condition is applied if we choose SpecularReflection model. When a particle strikes a perfect thermal wall at temperature Tw, all three components of the velocity are reset according to a biased Maxwellian distribiution. For MixedDiffuseSpecular we need to specify the accomodation coeficient.

Binary Collision Model chosen is Variable Hard sphere along with Larsen Bornakke Method. Other available options include Variable Hard sphere(without Larsnen Borgnakke) and NoBinaryCollision.

The values of Tref and relaxation collision number has been taken the default value. Although they can be varied their variation does not affect the simulation much.

Inflow boundary model has been chosen as freestream . The other option is no Inflow. In the freestream model particles are initialised at all the boundary pathches and are moved according to the boundary conditions after each time step.

After that the number density of freestream condition is specified. Later the molecular properties of all the constituent gases are specified. The molecular diameter is calculated according to the VHS model. The viscosity coefficient is a function of temperature.

A.2 DsmcInitialiseDict

```
numberDensities
{
   N2        0.777e16;
   O2        0.223e16;
};


temperature    300;

velocity       (10 0 0);
```

Here the number density of the initialised particle as well as their state is specified. Please note that for our case we do not want a lot of particles in the beginning. Thus the number density is kept low. The temperature and velocity is kept low so that the particles initilised are blown away by the particles coming from the nozzle(supersonic hyperbolic solution).

Appendix B : Boundary Fields

Boundary Fields are specified in the 0 folder of the case directory and are listed below:

- **boundaryT, boundaryU:**

Perhaps the most important files of the 0 folder. The wall and free stream conditions at the boundary are specified for velocity and temperature with these fields - only the data on the patches is used, the cell data is not. These are the only two fields which supply data to the case.

- **dsmcRhoN:**

The population of dsmc particles in cells is recorded to visualise how well the cell population conditions required for dsmc are met.The boundary conditions are zeroGradient because only cell data is meaningful.

- **fD, q:**

The wall heat flux (q) and force density (fD, i.e. stress vector) is recorded with these fields - only the data on wall patches is relevant, the cell data is not.

- **iDof, internalE, linearKE, momentum, rhoM, rhoN:**

These fields are the densities of extensive quantities in the simulation, i.e. of number, mass, momentum, energy. Cell data is recorded in the internal field and the boundaryField is used to record the data of particles that strike wall patches. The properties of particles striking wall faces are weighted by 1/(Un*fA), where Un is the normal component of the particle's velocity and fA is the face area. This is done so that when intensive quantities, such as velocity or temperature, are evaluated on the wall the values are correct this allows velocity slip and temperature jump to be evaluated.

Appendix C : Good DSMC Practices

The accuracy of DSMC simulation relies principally on four main constraints.

- The computational cell size must be smaller than the local mean path if possible collision partners are restricted to a particle's current cell which how dsmcFoam operates. For most dsmc simulation it is prescribed to have the cell size less than one-third of the mean free path. Obviously if the cell size is greater than the chances of collision are less.
- The simulation time step must be chosen so that particles only cross a fraction of the average cell length in each time step, and the time step must also be smaller than the local mean collision time.
- The number of particles per cell must be large enough to preserve collision statistics. Good dsmc practice suggests using a minimum of 10 particles in each cell for 2D simulation and 30 particles for 3D simulations.
- The statistical scatter is determined by the number of samples, and for the steady state problems sampling must not be started until a sufficient transient period has elapsed.

Appendix D : Command Line

The following commands must be executed in the following order for a simulation in dsmcFoam.

dsmcInitialise – This command is a pre processing utility to create initial configurations of DSMC particles in arbitrary geometries. DsmcInitialiseDict can be modiefied by entering proper number density and initial velocity and temperature.

decomposePar – This command is used to decompose the domain for parallel run of the program. DecomposeParDict can be modified to set domain decomposition parameters.

dsmcFoam - We can run the case in a single processor by running this command

mpirun -np 8 dsmcFoam – parallel - This can be used for running dsmc in 8 processors in parallel. Please note that your domain must be decomposed in 8 processors.

mpirun --hostfile machines.tbl -np 8 dsmcFoam -parallel - This can be used to run the dsmc simulation in nodes other that your local computer. For this you need to create a file named machines.tbl . Mention the nodes that you want your simulation to run in. Exmaple

```
node211
node214
```

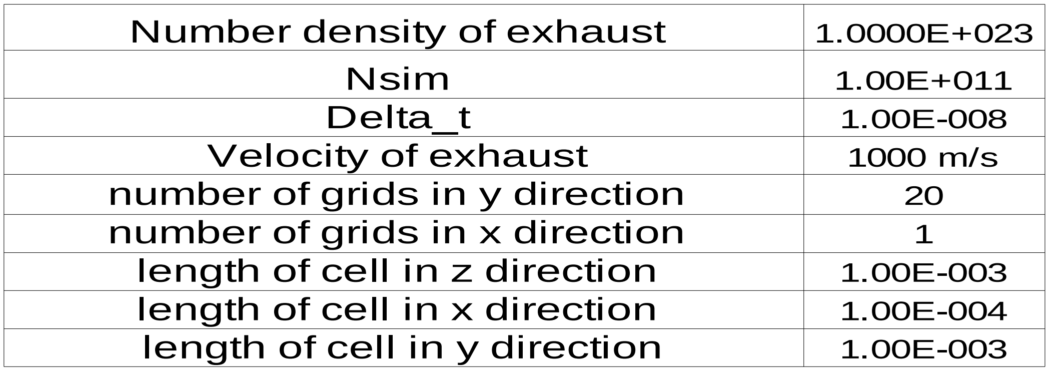
reconstructPar – helps to reconstruct the decomposed domains to single domain. The results before reconstruction are stored in different processor directories. After running this program the results would be combined and would show in the case directory.

paraFoam - This would open your simulation results in the paraview software for post processing .

Appendix E: Useful calculation

Since the accuracy of the dsmc depends highly on the number of particles per cell it is mandatory for us to make sure that number of particles entering the domain through rocket nozzle is just enough.

We will understand this with an example



Number density of exhaust	1.0000E+023
Nsim	1.00E+011
Delta_t	1.00E-008
Velocity of exhaust	1000 m/s
number of grids in y direction	20
number of grids in x direction	1
length of cell in z direction	1.00E-003
length of cell in x direction	1.00E-004
length of cell in y direction	1.00E-003

$$(NumberOfParticlesPerCell) * \frac{TotalNumberOfCell * Nsim}{VolumeOfGrid} = NumberDensity$$

$$\text{volume} = \mathrm{L}_y * L_z * V_x * delta_t$$

$$\text{volume} = 20\text{x}10^{-3} * 1\text{x}10^{-3} * 1000 * 1\text{x}10^{-8}$$

$$10^{11} * \frac{Numberofparticlespercell * (20 * 1)}{2\text{x}10^{-10}} = 10^{23}$$

Thus number of particles per cell comes out to be 10. Distance travelled in one time in x direction in one time step would be 1000 * 1e-8 ,i.e , 1e-5 m . Thus, the particles would take 10 time steps to cross the cell in x direction total number of particles would come out to be 100 .

References

- G.A Bird , *Molecular Gas Dynamics* , Oxford University Press, Oxord,1976.
- G.A. Bird , Molecular Gas Dynamics and Direct Simulation of Gas Flows, Clarendon Press , Oxford Science Publications , 1994.
- www.gab.au.com
- Christopher Kannenberg , *Computational Methods for the Direct Simulation Monte Carlo Technique with Application to the Plume Impingement.*
- Dr . Anil Lal *, ISRO Technical Respond Program . On DSMC Simulation.*
- *OpenFOAM user guide, 2.3.0.*