

2012

Immersed boundary method for Boltzmann model kinetic equations

Cem Pekardan

Sruti Chigullapalli
Purdue University

Alina A. Alexeenko
Purdue University - Main Campus, alexeenk@purdue.edu

Follow this and additional works at: <http://docs.lib.purdue.edu/aaepubs>



Part of the [Engineering Commons](#)

Recommended Citation

Pekardan, Cem; Chigullapalli, Sruti; and Alexeenko, Alina A., "Immersed boundary method for Boltzmann model kinetic equations" (2012). *School of Aeronautics and Astronautics Faculty Publications*. Paper 47.
<http://dx.doi.org/10.1063/1.4769542>

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

Immersed Boundary Method for Boltzmann Model Kinetic Equations

Cem Pekardan*, Sruti Chigullapalli*, Lin Sun[†] and Alina Alexeenko*

**School of Aeronautics and Astronautics, Purdue University*

[†]NNSA Center for Prediction of Reliability, Integrity and Survivability of Microsystems (PRISM) and Rosen Center for Advanced Computing, West Lafayette, IN 47907 USA

Abstract. Three different immersed boundary method formulations are presented for Boltzmann model kinetic equations such as Bhatnagar-Gross-Krook (BGK) and Ellipsoidal statistical Bhatnagar-Gross-Krook (ESBGK) model equations. 1D unsteady IBM solution for a moving piston is compared with the DSMC results and 2D quasi-steady microscale gas damping solutions are verified by a conformal finite volume method solver. Transient analysis for a sinusoidally moving beam is also carried out for the different pressure conditions (1 atm, 0.1 atm and 0.01 atm) corresponding to $Kn=0.05, 0.5$ and 5. Interrelaxation method (Method 2) is shown to provide a faster convergence as compared to the traditional interpolation scheme used in continuum IBM formulations. Unsteady damping in rarefied regime is characterized by a significant phase-lag which is not captured by quasi-steady approximations.

Keywords: Boltzmann Equation, ESBGK, Immersed Boundary Method, Gas Damping, MEMS, Rarefied Gas Dynamics

PACS: 47.61.Fg, 47.45.Ab, 47.10.A-, 47.11.Df

INTRODUCTION

Unsteady rarefied gas flows interacting with elastic structures arise in a number of applications, e.g. large-displacement motion of microbeams and membranes in radio-frequency MEMS switches and filters. The direct simulation Monte Carlo (DSMC) method has been widely applied to compute high-speed rarefied gas flows for fixed solid boundaries. A formulation of moving boundary DSMC method has been presented recently by Rader et al. [1]. Since the cost of stochastic DSMC simulations increases for low-speed flows, the deterministic solution of Boltzmann model kinetic equations may provide a useful framework for unsteady rarefied flows in MEMS and other low-speed applications.

Fluid-structure interaction simulation with either partial or complete remeshing is algorithmically complex and encounters additional numerical errors due to mesh skewness and interpolation between meshes [2]. Fixed-grid methods such as the immersed boundary method are increasingly being applied in continuum CFD to handle complex moving geometries. This paper presents the immersed boundary method formulation for the Boltzmann model kinetic equation with the ESBGK and BGK model collision operators.

Immersed boundary method involves discretizing the solid and fluid domains separately where solid mesh is immersed in the background mesh and moves through it. Cells in the background mesh are marked depending on the location of the solid mesh (solid cells, fluid cells and immersed boundary (IB) cells). Flow field is discretized and solved for each velocity direction at fluid cells and boundary conditions are applied at the IB faces (faces between IB cells and fluid cells) for the incoming directions to the flow. There are three immersed boundary schemes developed in this study to calculate boundary conditions on the solid body and to find the boundary conditions that are applied on the IB faces. The first scheme involves interpolation of velocity distribution function from fluid cells to the solid boundary faces using a least square method. Conservation of mass flux is applied at the solid boundary faces with a consequent interpolation back to the IB faces and these are used as the Dirichlet boundary condition for the finite volume solution of ESBGK equations. The second scheme involves interpolation of macroparameters to IB faces and then computation of velocity distribution function at solid faces through a space-homogeneous relaxation operator. After applying conservation of mass flux, distribution function is relaxed back with the same equation and used as a Dirichlet condition for all incoming directions. Third method (Relaxation) only uses relaxation scheme to calculate the boundary values. For the first test, 1D moving piston problem is considered. Temperature and pressure distributions for several step sizes are compared with DSMC results of [1]. 2D steady results are compared with the conformal fixed boundary 2D results. Furthermore, a sinusoidally moving beam is analyzed with Interrelaxation method and transient damping force values are presented.

IMMERSED BOUNDARY METHOD FORMULATIONS

ESBGK and BGK type Boltzmann equations are solved numerically employing a finite volume method (FVM) with a second order quadrant-splitting scheme applied in physical space. Details of the numerical method for FVM (steady-conformal ESBGK method without IBM) that are used to verify 2D steady IBM-ESBGK results can be found in [5, 6].

Domain Marking Procedure

In the marking procedure, solid body is divided into a combination of particles whose positions are determined on the background mesh using an octree search described in [7]. Cells on the background mesh are marked depending on the number of material point they have at the given instant of time. All cells having no material points are marked as fluid cells, whereas the boundary cells having at least one material points and have at least one neighbor of fluid cell is marked as Immersed Boundary (IB) Cells and all the other cells are marked as solid cells.

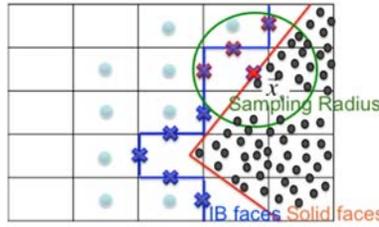


FIGURE 1. Marking procedure representation.

For the interpolations used in the formulations in this paper, least squares method is implemented. As described in [2], the interpolating polynomial is assumed to be of the form,

$$\phi_{linear} = \beta_0 + \beta_1 x, \quad \phi_{2nd\ order} = \beta_0 + \beta_1 x + \beta_2 x^2. \quad (1)$$

The values of ϕ reconstructed at n neighboring cell centers and solid faces is given by $M\beta$. For linear and second order interpolation,

$$M_{linear} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}, \quad M_{2nd\ order} = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}.$$

Assume n is greater than the number of coefficients m in both cases, coefficients $M\beta_m$ are found by minimizing the distance between re-constructed values $M\beta$ and actual values of ϕ . For example, for first order interpolation given cell centroids x_i and functional values $M\phi_i$, the interpolation function $\phi_{interpolant} = \beta_0 + \beta_1 x$ can be found from minimizing $R^2 = \sum_{i=1}^n [\phi_i - (\beta_0 + \beta_1 x_i)]^2$.

$$\frac{\partial R^2}{\partial \beta_0} = 0, \quad \frac{\partial R^2}{\partial \beta_1} = 0. \quad (2)$$

Writing in matrix form,

$$\begin{bmatrix} \sum 1 & \sum x_i \\ \sum x_i^2 & \sum x_i \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \sum \phi_i \\ \sum x_i \phi_i \end{bmatrix}, \quad \begin{bmatrix} \sum 1 & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} \sum \phi_i \\ \sum x_i \phi_i \\ \sum x_i^2 \phi_i \end{bmatrix}.$$

In compact form, $R^2 = |\phi - M\beta|^2$ is minimized by,

$$\beta = (M^T M)^{-1} M^T \phi. \quad (3)$$

Solution Procedure

- Initialize the macroparameters, distribution function and equilibrium velocity distribution function values (depending on BGK or ESBGK), and initialize positions and velocities of material points as described above.
- Mark the cells and faces on the background mesh as described in the Domain Marking Procedure section.
- For the given time step, determine wall boundary conditions on the velocity distribution function for each direction at the immersed boundary faces on the background mesh using one of the three procedures described in the following subsection.
- Discretize and solve the velocity distribution function values at the fluid cells.
- Iterate last 3 steps in order to achieve convergence for the given time step.
- Determine the new locations of the material points for the next time step and go to the second step until the desired time steps are completed.

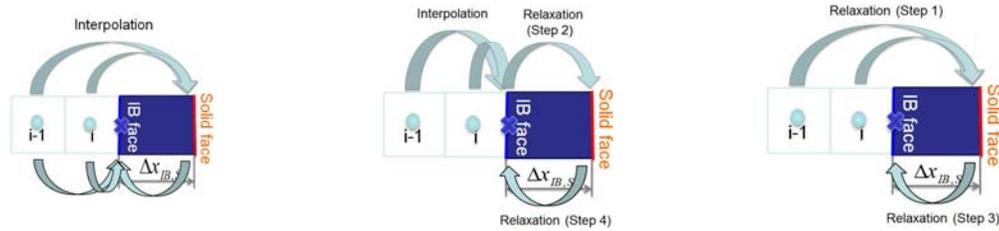


FIGURE 2. Procedure for calculation of IB face incoming velocity distribution function values for IBM-Interpolation (Left), IB-Interrelaxation (Middle), IB-Relaxation (Right) Methods.

Determination of Wall Boundary Conditions for Velocity Distribution Function - Interpolation Method

The linear and second order least squares method is used to implement boundary conditions at the solid faces and use these values at the IB faces to apply as a Dirichlet boundary condition for the linear system. The solution procedure can be described as the following steps:

Step 1: Interpolate the distribution function values from fluid cells in the sampling radius to solid faces which are the boundaries of the solid body.

Step 2: The boundary condition is implemented as a normal diffuse wall which involves the conservation of mass flux at the face of the solid at temperature T_{wall} and number density n_{wall} . Starting from mass flux conservation equation at the wall,

$$\sum_{\vec{c} \cdot \vec{n}_s > 0} \vec{c} \cdot \vec{n}_s f_{s,out} + \sum_{\vec{c} \cdot \vec{n}_s < 0} \vec{c} \cdot \vec{n}_s f_{s,in} = 0, \quad (4)$$

$$\sum_{\vec{c} \cdot \vec{n}_s > 0} \vec{c} \cdot \vec{n}_s f_{s,out} + \sum_{\vec{c} \cdot \vec{n}_s < 0} n_{wall} \exp\left(\frac{-(\vec{c} - \vec{V})^2}{T_{wall}}\right) = 0, \quad (5)$$

$$n_{wall} = \frac{-\sum_{\vec{c} \cdot \vec{n}_s > 0} \vec{c} \cdot \vec{n}_s f_{s,out}}{\sum_{\vec{c} \cdot \vec{n}_s > 0} \vec{c} \cdot \vec{n}_s \exp\left(\frac{-(\vec{c} - \vec{V})^2}{T_{wall}}\right)}, \quad (6)$$

where the velocity distribution function incoming to the flow is written as,

$$f_s = n_{wall} \exp\left(\frac{-(\vec{c} - \vec{V})^2}{T_{wall}}\right). \quad (7)$$

Step 3: The function values are interpolated back to the IB faces using the interpolation scheme described above using both fluid cells and solid faces that diffuse wall boundary condition is implemented.

Step 4: For the incoming directions at the IB faces, Dirichlet boundary condition is implemented for the discretization and solution of the flow field.

Determination of Wall Boundary Conditions for Velocity Distribution Function - Interrelaxation Method

Interrelaxation formulation implements interpolation of the collision frequency and velocity distribution function to the IB faces. Then the solid face values are found using a relaxation procedure defined below.

Step 1: Distribution function and equilibrium distribution function values are interpolated from only fluid cells to IB faces. Collision frequency is interpolated from solid faces and fluid cells in the sampling radius to the IB faces.

Step 2: The outgoing function values from the fluid cells are calculated using the relaxation procedure to the local equilibrium distribution functions (f_γ), which is the solution of the differential equation [8, 9],

$$\frac{Df}{Dt} + (f - f_\gamma) = 0. \quad (8)$$

which has the following solution,

$$f_{s,out} = f_{\gamma,IB} + (f_{IB,out} - f_{\gamma,IB}) \exp\left(\frac{-t}{\tau_{IB}}\right). \quad (9)$$

Step 3: Conservation of mass flux algorithm is applied as described in the interpolation part of the section to determine the incoming velocity distribution function values.

Step 4: All the incoming $f_{IB,in}$ at the IB face are then calculated from relaxation of $f_{s,in}$ in each direction using,

$$f_{IB,in} = f_{\gamma,IB} + (f_{s,in} - f_{\gamma,IB}) \exp\left(\frac{-t}{\tau_{IB}}\right). \quad (10)$$

Determination of Wall Boundary Conditions for Velocity Distribution Function - Relaxation Method

Instead of using least squares interpolation, this method uses a relaxation formulation to compute solid face distribution function values and apply boundary conditions. The IB face values to be implemented as boundary conditions are calculated by the same relaxation equation using the solid face values at the sampling radius.

Step 1: Using a similar relaxation formula described above, solid face distribution values are determined for the incoming direction (relative to the solid faces) distribution function values,

$$f_{solid} = f_{\gamma,fluidcells} + (f_{fluidcells} - f_{\gamma,fluidcells}) \exp\left(\frac{-t}{\tau_{fluidcells}}\right). \quad (11)$$

Step 2: Conservation of mass flux algorithm is applied as described in the interpolation part of the section to determine the incoming velocity distribution function values.

Step 3: Only equilibrium distribution function is interpolated to solid faces for relaxation from solid faces to the IB faces for the incoming directions to the flow. Collision frequency at the walls is calculated using power law for both Relaxation and Interrelaxation methods.

$$f_{IB,in} = f_{\gamma,solid} + (f_{s,in} - f_{\gamma,solid}) \exp\left(\frac{-t}{\tau_{solid}}\right). \quad (12)$$

In the 2D implementation, if a given IB cell has more than one IB faces (as in the corner parts), averaging of the IB face values are done before imposing the value in the ghost cell as a Dirichlet boundary condition in the incoming directions of distribution function. For the relaxation parts of the methods in 2D implementation, a linear summation based on the distance between two points where the relaxation is done in sampling radius, is implemented to have an only one value at the solid faces to calculate the boundary conditions.

RESULTS

For the first test, immersed boundary method is implemented for 1D moving piston case, which was investigated with DSMC moving boundary algorithms [1] at pressure 266.633 Pa and temperature 273.15 K. The same domain is generated using a 1D user defined code, in which the piston is located at the midpoint of a domain (500 μ m) and has a thickness of 20 μ m. Properties of argon in Bird [10] are used for the comparison with the DSMC results. Domain

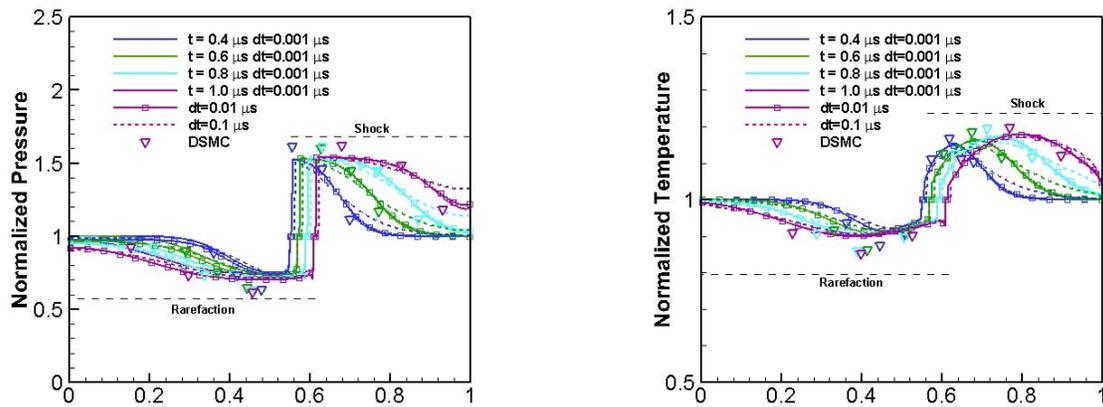


FIGURE 3. Pressure (Left) and Temperature (Right) distributions over the domain at different time instants with IB-ESBGK Interrelaxation method and DSMC [1].

is discretized into 100 cells and immersed boundary formulation is applied to the moving boundaries for the left and right of the domain.

Temperature and pressure distributions along the domain at different time instants are compared with the DSMC results using different time steps sizes. When the time step size decrease, results agree with the DSMC results. Shock wave generation at the compression part of the domain and rarefaction wave generation at the expansion part are both captured with IB-ESBGK results as well.

A 2D test case is done for the Immersed Boundary ESBGK methods described above. Stress values at the wall boundary are calculated by least squares interpolation of local stress values from the neighboring fluid cells. Domain marking is done by the procedure described in previous parts and represented in Figure 4.

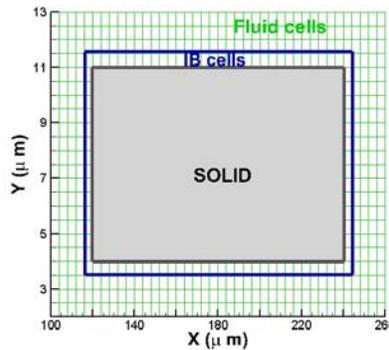


FIGURE 4. Marking for the 2D test case.

An $7\mu\text{m}$ by $120\mu\text{m}$ beam is placed onto a $360\mu\text{m}$ by $100\mu\text{m}$ fluid domain and $+2\text{ m/s}$ vertical velocity is given in 1 atm pressure environment for a steady case. The results of the given methods are compared with the conformal ESBGK solver without IBM [6].

Damping force values created on the beam per unit length can be seen in Figure 8-right for a background mesh with 10,800 elements (Cell $dx=2\mu\text{m}$) and 21,600 elements (Cell $dx=1\mu\text{m}$). Furthermore, it is seen that when the mesh gets finer, all the methods approach to the conformal mesh values, whereas the Interrelaxation method gives an error less than 2 percent for the fine mesh, which verifies the method.

Pressure distributions at the top and the bottom of the wall are also compared with the presented methods in Figure 8-left and it is seen that Interrelaxation method gives the best results which was expected from the converged force values.

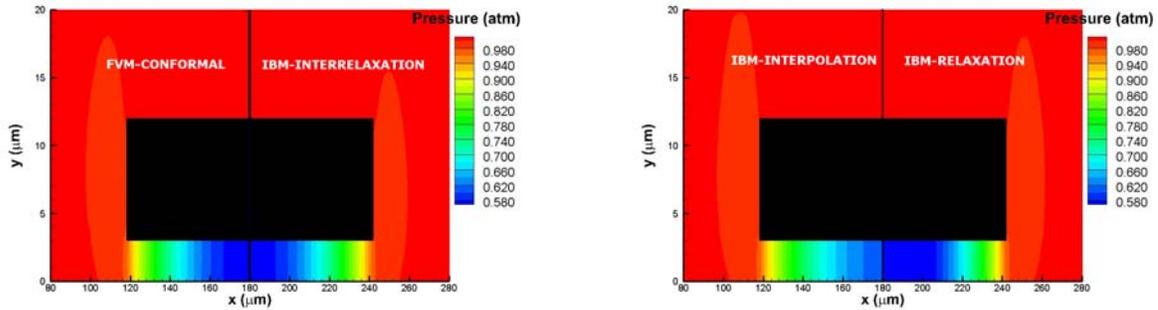


FIGURE 5. Pressure contours for the FVM-Conformal and IB-ESBGK methods for 2D steady case.

Figure 6 and 7 shows that there is a slight deviation from the conformal results for the velocities due to the mesh resolution and interpolation errors accumulated during the iterations.

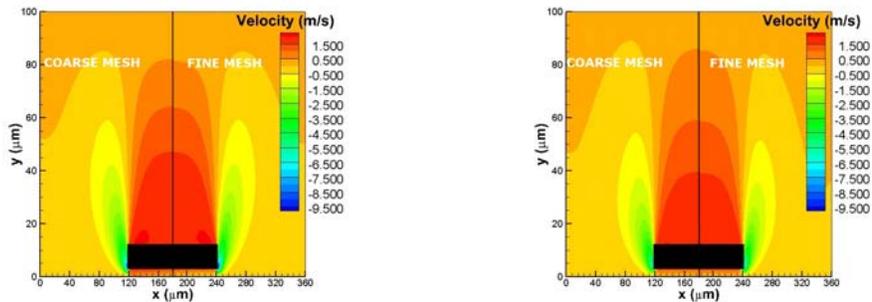


FIGURE 6. Velocity contours for FVM-Conformal (left) and IB-Interpolation (right) methods for 2D steady case with 10,800 (Coarse) and 21,600 (Fine) mesh elements.

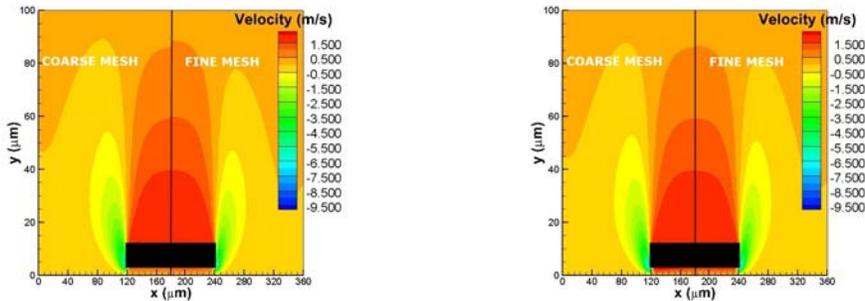


FIGURE 7. Velocity contours for IB-Relaxation (left) and IB-Interrelaxation (right) methods for 2D steady case with 10,800 (Coarse) and 21,600 (Fine) mesh elements.

For a sinusoidally moving beam, transient analyses are done and pressure contours are presented with the IB-Interrelaxation method, and snapshots can be seen at figures 10 and 11. A time step size of $0.25 \mu\text{s}$ and 10,800 mesh elements are used.

For the coarser mesh, the distance between solid face and IB affects the results because flow on the surface is not well resolved. When the mesh gets finer, this effect is reduced and does not change the overall result. For the solution of the complex geometries with immersed boundary method requires enough resolution around the complex parts for the complex flow pattern. Therefore, automatically the distance effect is eliminated.

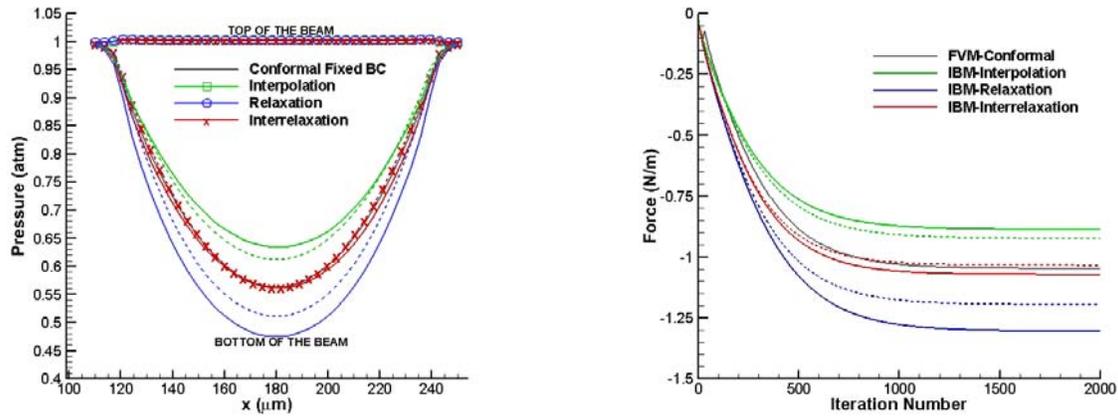


FIGURE 8. Pressure distributions at the top and bottom of the beam (Left) and steady damping force comparison (Right) for different methods for coarse (Solid Line) and fine (Dashed Line) meshes.

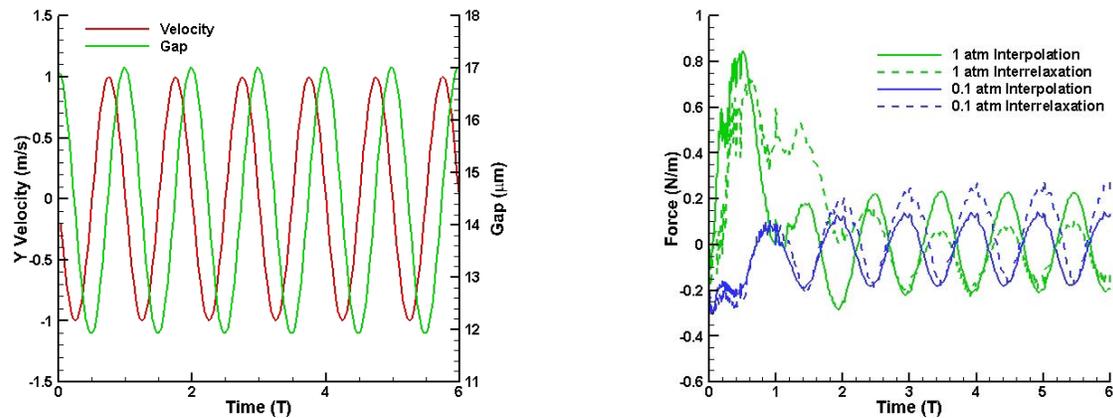


FIGURE 9. Velocity and gap propagation over time (Left) and unsteady damping force analyses for 1 atm and 0.1 atm initial pressure conditions (Right).

CONCLUSION

Three different immersed boundary formulations for Boltzmann Model Kinetic Equations are presented in this manuscript. Interpolation method uses a least squares interpolation method to apply the boundary condition at the solid faces, whereas, relaxation method uses a relaxation scheme which is relaxation to local equilibrium functions for each cell in the sampling radius. On the other hand, Interrelaxation method combines both methods which decreases the errors due to the values of equilibrium function in the period of relaxation.

First of all, 1D verification with the DSMC results was done and it was shown that for the different instants of time, the same results are produced for both temperature and pressure profiles for the small time step sizes. Results start to deviate from the DSMC results when time step is greater than 0.01 because of the discretization errors. However, it is possible to increase the DSMC step size which was 0.001, from 0.001 to 0.01 without changing the results which decrease the CPU time significantly.

For the comparison of FVM-Conformal results with IBM methods, damping force values are calculated interpolating

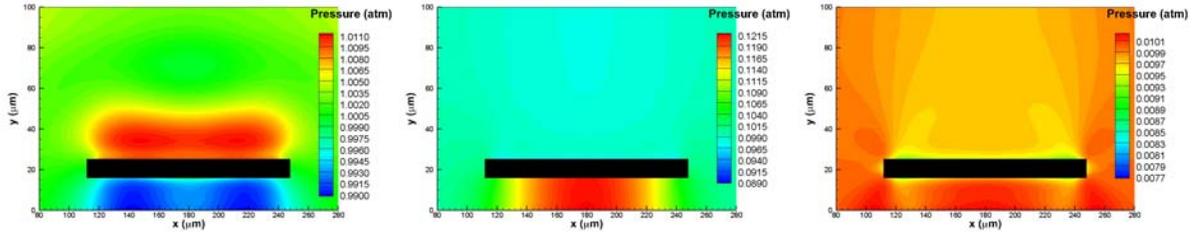


FIGURE 10. Unsteady pressure contours at 3 periods ($48\mu\text{s}$) for 1 atm (Left), 0.1 atm (Middle) and 0.01 atm (Right) initial pressure conditions.

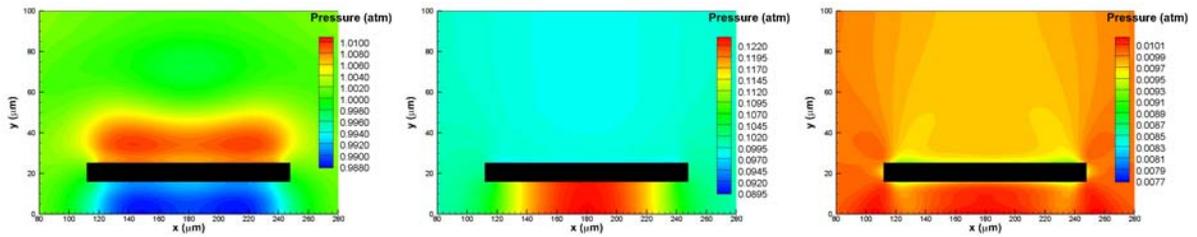


FIGURE 11. Unsteady pressure contours at 4 periods ($64\mu\text{s}$) for 1 atm (Left), 0.1 atm (Middle) and 0.01 atm (Right) initial pressure conditions.

the stress values to the solid faces. It was shown that Interrelaxation method decreased the error to around 4 percent when a finer mesh is used. On the other hand, interpolation and relaxation methods approached the conformal FVM results by 7 percent for the same mesh sizes.

In figures 10 and 11, it was seen that for 0.1 atm initial pressure conditions (transition Knudsen number), there is a phase lag in the transient pressure contours and damping force.

ACKNOWLEDGMENTS

This work has been supported by NNSA Center for Prediction of Reliability, Integrity and Survivability of Microsystems at Purdue University under Contract Number DE-FC52-08NA28617

REFERENCES

1. D. J. Rader, M. A. Gallis, and J. R. Torczynski, *AIP Conf. Proc.* **1333**, 760-765 (2010).
2. L. Sun, S. Mathur and J. Murthy, *Numer. Heat. Tr. B-Fund.* **58**, 217-241 (2010).
3. R. Mittal, G. Iaccarino, *Annu. Rev. Fluid Mech.* **37**, 239-261 (2005).
4. Y. Tseng, J. Ferziger, *J. Comput. Phys.* **192**, 593-623 (2003).
5. L. Holway Jr, *Phys. Fluids* **9**, 1958 (1966).
6. A. Alexeenko, S. Gimelshein, E. Muntz, and A. Ketsdever, *Int. J. Therm. Sci.* **45**, 1045-1051 (2006).
7. R. L. Lohner, *Commun. Appl. Numer. Meth.* **4**, 123-135 (1998).
8. C. K. Chu, *Phys. Fluids* **8**, 12-22 (1965).
9. Y. Ruan and A. Jameson, *AIAA* **0550** (2002).
10. G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford: Clarendon (1994).