JOURNAL OF THERMOPHYSICS AND HEAT TRANSFER
Vol. 14, No. 3, July–September 2000

Direct Simulation Methods for Low-Speed Microchannel Flows

Chunpei Cai,* Iain D. Boyd,† and Jing Fan‡
Cornell University, Ithaca, New York 14853
and
Graham V. Candler§
University of Minnesota, Minneapolis, Minnesota 55455

Large statistical scatter and effective pressure boundary conditions are two critical problems in the computation of microchannel flows with the direct simulation Monte Carlo (DSMC) method. To address these issues, an extension of the DSMC-IP (information preservation) coupled method is developed from the one-dimensional case to the two-dimensional case for microchannel flow. Simulation results in a microchannel flow from DSMC, IP, and numerical and analytical solutions to the Navier–Stokes equations are compared. The DSMC-IP coupled method successfully reduces the large statistical scatter usually obtained with DSMC in such low-speed flow systems. It also provides a suitable implementation of pressure boundary conditions.

Introduction

MICROCHANNELS are an important component of many microelectrical mechanical systems (MEMS).[1] Successful numerical simulation of the flow field inside these devices is required to understand small scale flow phenomena. Adopting a standard computational fluid dynamics (CFD) technique is not appropriate because CFD is based on the continuum assumption, which is only good for the continuum regime ($Kn < 0.001$), and acceptable for the temperature jump and velocity slip region ($0.001 < Kn < 0.1$) if a slip wall condition is adopted instead of nonslip boundary conditions. However, for microchannel flows under experimental conditions,[1–4] the flows are sometimes in the transition regime ($0.1 < Kn < 10$). Here, rarefied effects are significant, and CFD methods are not reliable. The direct simulation Monte Carlo method (DSMC)[5] is accurate for all flow regimes because it is based on kinetic theory and does not rely on the continuum assumption. Many researchers have already performed much work on simulation of microchannel flow with the DSMC method.[6–10] However, there are still many difficulties, and in some researchers' belief[8] it is impossible to use DSMC to simulate microchannel flows under experimental conditions. Indeed, there are many experimental results, but no corresponding DSMC simulations are reported yet. To statistically simulate the flow under experimental conditions, we must overcome two particular difficulties: statistical scatter and proper implementation of pressure boundary conditions. In this study we will discuss these difficulties in detail. A new technique, the DSMC-IP (information presentation) coupled method, is presented to address these problems. Two cases are computed in the present study: a simplified test case and a case under experimental conditions.

Difficulties Associated with DSMC Calculation

Statistical Scatter

Usually, the flow velocity in microchannels under typical experimental conditions is very low. For example, in the experiments of Pong et al.,[2] the inlet velocity is about 20 cm/s. If we suppose the velocity obeys a Maxwellian distribution, then at room temperature for nitrogen, the standard deviation is $\sigma = \sqrt{(2RT)} = 422$ m/s. If we suppose the sampling processes are totally independent from step to step, then the statistical scatter in the final DSMC result will be $\sigma' = \sigma/\sqrt{N}$, where N is the sample size, which is proportional to the number of sampling time steps. If we consider the ratio of noise to signal $\sigma'/S = \frac{1}{3}$ is enough for the signal to be clear from the DSMC simulation result, then we need a sampling size of $N = (3\sigma/S)^2 = 4.0 \times 10^7$. If we use 1000 particles in each cell, then we need 40,000 time steps to get a reasonable value of the mean velocity. In the DSMC technique we have to adopt a grid with very small cells, which results in a large number of cells, so it is almost too expensive to use DSMC for real microchannel flows. This is one of the main reasons why the usual approach is to solve the Navier–Stokes equations with slip boundary conditions rather than use the DSMC method, even though the Navier–Stokes equations are less physically accurate under rarefied conditions. Previous DSMC simulations of microchannel flows concentrated on channels with small geometry (usually with a length of several mean free paths[6,7]) or large stream velocities.[8,9] Unfortunately, these are not the real flow conditions in the experiments. Simulation of microchannel flow with the standard DSMC method requires a tremendous amount of computational power unless we can successfully reduce the statistical scatter.

Inlet and Outlet Pressure Boundary Conditions

For supersonic flow the boundary conditions are straightforward for the DSMC method. Particles are simply removed when they cross inlet or outlet boundaries, and a certain number of particles are injected into the cells at inlet and outlet according to a Maxwellian distribution, provided the velocity, temperature, and number density at these locations are known. However, for microchannels the flow is low subsonic. Also, because the domain is too small to allow measurement by common diagnostics, experiments cannot provide the velocity data at the channel inlet and outlet. Instead, only the total flux rate and pressures can be measured. In this case adopting a pressure boundary condition that does not require information about the velocity magnitude is required for the DSMC method. Several types of DSMC pressure boundary condition implementations have been proposed.[6,7] We review their implementation and then discuss why those methods do not work.

Iterative Flux Method

The following method based on a Maxwellian distribution was proposed by Ikegawa and Koboyashi.[6] For the flow of mean velocity $(\pm U, 0, 0)$, the probability density function $f_{\text{in}}(C)$ of velocity

Received 11 October 1999; revision received 28 February 2000; accepted for publication 14 March 2000.

*Graduate Research Assistant, School of Mechanical and Aerospace Engineering.
†Associate Professor, School of Mechanical and Aerospace Engineering. Senior Member AIAA.
‡Research Associate, School of Mechanical and Aerospace Engineering. Member AIAA.
§Associate Professor, Aerospace Engineering and Mechanics Department. Senior Member AIAA.

$C(u, v, w)$ for molecules entering the flowfield from an upstream or downstream boundary (across a surface normal to the flow) is

$$f_{\text{in}}(C) = \left(2u\middle|\ \pi C_m^4 M\right)\exp\left\{\left(-1\middle|\ C_m^2\right)[(u \pm U)^2 + v^2 + w^2]\right\} \quad (1)$$

where

$$M = \exp\left(-U^2\middle|\ C_m^2\right) \pm \sqrt{\pi}(U/C_m)[1 \pm \text{erf}(U/C_m)] \quad (2)$$

the plus or minus sign is for the upstream ($u + U > 0$) or downstream ($u - U) < 0$ case, respectively. C_m is the $\sqrt{(2RT)}$ the most probable molecular speed; R is the specific gas constant; T is the temperature of gas in the upstream or downstream; and erf is the error function.

The number of entering molecules across an area A from the upstream or downstream boundaries is given by the following equation:

$$N_{\text{in}} = \left(nC_m M/2\sqrt{\pi}\right)\delta t A \quad (3)$$

where n is the number density in the upstream or downstream ($=P/kT$), P is the pressure in the upstream or downstream, and k is the Boltzmann constant.

In Ref. 6, the proposal is made to estimate the mean flow velocities $U(t)$ at the boundaries based on the flux of molecules flowing in $N_{\text{in}}^{(t-\delta t)}$ and out $N_{\text{out}}^{(t-\delta t)}$ one time step from the upstream or downstream boundaries:

$$U(t) = \frac{N_{\text{in}}^{(t-\delta t)} - N_{\text{out}}^{(t-\delta t)}}{n\delta t A} \quad (4)$$

In this equation the product of the left-hand side and the denominator of the right-hand side, that is, $(Un\delta tA)$, should be regarded as the flux of molecules passing through the boundaries. Because $(n\delta tA)$ is a small number and $(N_{\text{in}} - N_{\text{out}})$ is also a small number, there can be large numerical scatter in the values of $U(t)$.

Characteristic Line Method

A microchannel flow simulation with pressure boundary condition was made by Nance et al.[7] At the inlet a method very similar to that of Ref. 6 was adopted. At the outlet properties are controlled by the characteristic line method as follows:

$$(\rho_e)_m = \rho_m + (P_e - P_m)\left|\ a_m^2\right. \quad (5)$$

$$(U_e)_m = U_m + (P_m - P_e)/\rho_m a_m \quad (6)$$

$$(V_e)_m = V_m \quad (7)$$

$$(T_e)_m = P_e/[(\rho_e)_m R] \quad (8)$$

where subscript e represents an exit quantity, m represents an instantaneous computational result, e.g., P_e represents the given pressure, P_m is the computed pressure, $(\rho_e)_m$ represents the density after correction, and a_m is the sonic speed.

Discussion

The essence of these methods is iteration. They require evaluation of velocity U or pressure P_m in a given cell every one or several time steps. Because of the statistical scatter in the DSMC values of velocity and number density, as just mentioned, these methods are too expensive and impractical to adopt in the DSMC method for very slow cases.

In practice, when a flow becomes steady, the velocity obtained by iteration methods consists of two parts:

$$U'(x, y, t) = U(x, y) + \tilde{u}(x, y, t) \quad (9)$$

where $U'(x, y, t)$ represents the velocity obtained by an iteration method at each time step in different inlet or outlet cells; $U(x, y)$ represents the real physical velocity at that location after the flow becomes steady; and $\tilde{u}(x, y, t)$ is the difference between these two values. We want $\tilde{u}(x, y, t)$ to be as small as possible, or at least, to share the same order as $U(x, y)$; otherwise there will be variation in the inlet or outlet boundary conditions. Because the flowfield is subsonic, this problematic boundary condition implementation will affect the whole flowfield.

To improve the implementation of pressure boundary conditions, scatter in the evaluation of mean velocities or pressure must be effectively suppressed. Common remedies are to average over several time steps or to average over several neighboring cells. These methods work to some degree in some cases but fail for extremely slow flow cases. Experience shows that in the implementation of Ref. 7, $\tilde{u}(x, y, t)$ is much larger than $U(x, y)$ even when we use a large number of particles in the inlet or outlet boundary cells. The preceding methods are therefore expensive and impractical. Some new implementation of a pressure boundary condition is required before we can successfully make DSMC simulations of microchannel flow under experimental conditions.

IP Method

Background

The IP method, first proposed by Fan and Shen,[11] was used to overcome the problem of statistical scatter in low-speed, constant density flow systems. It achieved great success for several one-dimensional cases as shown in Ref. 11, including Couette flow, Poiseuille flow, and the Rayleigh problem. The main idea is based on the following consideration. In the sampled results produced by the traditional DSMC method, there is the physical component, which we want to isolate, and a statistical scatter component, which we try to suppress. In the IP method efforts are made to preserve and update the physical signal using a macroscopic point of view. For two-dimensional problems such as microchannel flow, the number density changes significantly at different locations. The original IP model of Ref. 11 is too simple for this kind of flow. In the present study we develop a more universal preservation method. The key basic points of the IP method are summarized as follows.

1) Particles used in the simulation have two roles: one is microscopic, representing single molecules, and one is macroscopic, representing a group of molecules. Because each particle in a DSMC computation usually represents a much larger number of real molecules, this supposition is physically reasonable.

2) Each particle in the simulation contains the usual, microscopic DSMC information: a position vector, a velocity vector, and internal energy. In addition, each particle possesses preserved, macroscopic information for the IP method, which, in the implementation described, here consists only of a preserved velocity vector. The IP method also associates preserved information with the computational cells: number density and velocity vector. Macroscopic conservation laws will be applied to the macroscopic IP information.

3) The DSMC solution develops as usual, such that the IP steps do not interfere with the DSMC results. This maintains the validity of the DSMC method. IP uses DSMC as a carrier, e.g., during every collision or wall reflection IP does not have any effect on DSMC, and particles redistribute their microscopic translational velocities and internal energies as usual. At the same time, however, IP redistributes the preserved information by different rules.

In the DSMC method collision is the only way for particles to change their velocity if there is no force field (e.g., gravity). In the IP method collisions can change the particle's velocity, but it is not the only way. In IP the cell is considered as a black box from a macroscopic point of view. Attention is paid to the total mass of particles in the cell and the net pressure force acting on this cell, from which we can compute the acceleration by

$$\boldsymbol{a} = \boldsymbol{F}_{\text{net}}/m \quad (10)$$

where $\boldsymbol{F}_{\text{net}}$ is the net pressure force acting on this cell, m is the total particle mass in this cell, and $\boldsymbol{a}$ is the acceleration. All particles in this cell are assumed to share this acceleration. This approach disregards the specific particles in the cell unlike DSMC. The use of Eq. (10) appears to omit the effects of convection and shear stress. These are included implicitly by the motions and collisions of the particles. Figure 1 shows the ways velocities are updated in these

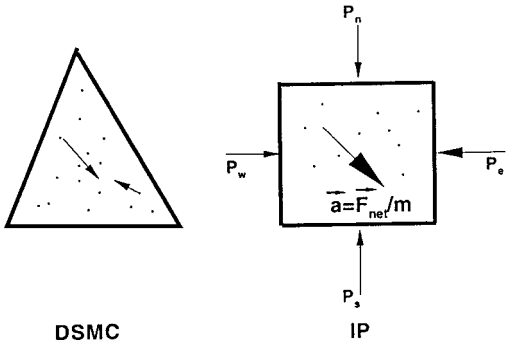


Fig. 1 Mechanisms for updating velocity in the DSMC and IP methods.

methods, where the subscripts e, w, s, n represent east, west, south, and north, respectively. In every time step we use the acceleration to change the preserved velocity of each particle.

In the paper by Fan and Shen,[11] only one-dimensional flow was considered, and so it was simply supposed that the flow is unidirectional with a given constant acceleration. For microchannel flow, which is two dimensional, to compute the acceleration is one of the main difficulties. We must calculate the acceleration, which requires knowledge of the pressure on all sides of each cell. In DSMC the scalar pressure is obtained from the thermal velocities:

$$p = \tfrac{1}{3}nm\left(\overline{c_1'^2} + \overline{c_2'^2} + \overline{c_3'^2}\right) \tag{11}$$

This equation does not work effectively for microchannel flows because of the scatter associated with the extremely low velocity and number density. Even supposing the flow is isothermal, we cannot use the preceding formula because the number density has significant statistical scatter. However, if we can further preserve the number density n in each cell, then by using the ideal gas law ($P = nmRT$) we can easily compute the pressure in each cell at each time step. The preserved number density for each cell will keep on changing from an initial flowfield until the flowfield becomes steady.

Consider the development from the original one-dimensional case to a two-dimensional implementation. In microchannel flow there is not much change in velocity from the inlet to the outlet, and so the translational temperature change can be neglected. For example, in the experiment of Ref. 2 the inlet velocity is about 20 cm/s. With a ratio of 2.5 between inlet and outlet pressures, the outlet velocity should be about 50 cm/s to maintain mass conservation. If we consider enthalpy is conserved for simplicity, then there will be a temperature difference between the inlet and the outlet $\Delta T = (V_{\text{out}}^2 - V_{\text{in}}^2)/2C_p = 0.0001$ K. So the isothermal assumption holds very well under low-speed conditions. We will see that this assumption is critical for the present DSMC-IP implementation.

In the present DSMC structure there are two main elements: particles and cells. In the original implementation of the IP method, three macroscopic velocity components are preserved for each particle. Therefore, there are 11 quantities associated with each particle, x, y, z, U, V, W, ϵ_{rot}, ϵ_{vib}, U_{ip}, V_{ip}, W_{ip}. The first eight quantities are for DSMC, representing position, velocity, internal energy; the last three are for IP to store the macroscopic velocity. In this study we continue with this implementation. One great improvement from the original one-dimensional IP method by Fan and Shen[11] to the present two-dimensional IP version is that information preservation is further extended to the cell level. Thus in every cell, number density n, macroscopic translational velocities U, V, W, and acceleration $\boldsymbol{a}$ are preserved. The reasons for introducing these quantities are for convenience in the sampling and in the net force computation in the IP method.

Main Steps

We couple the DSMC and IP methods together using the DSMC method as a carrier for the IP method. First let us review the main steps in the DSMC method:

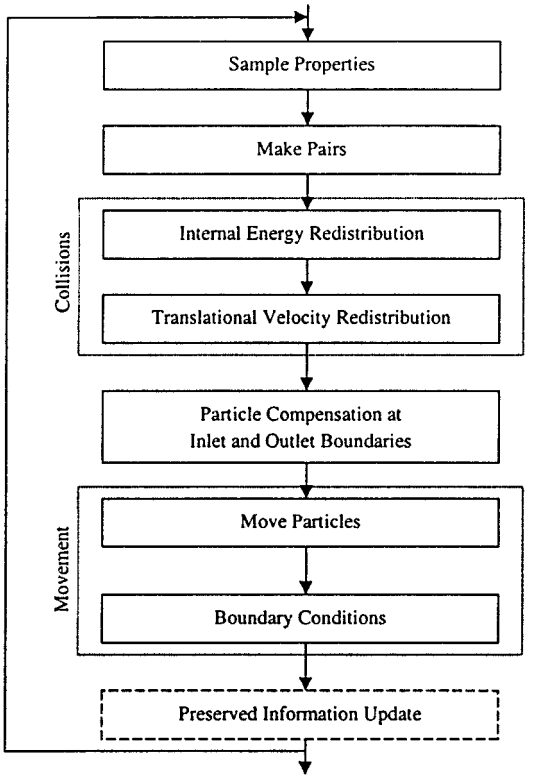


Fig. 2 Flowchart of core loop in the DSMC-IP method.

1) The time interval $[0, t]$, over which the solution is sought, is divided into subintervals with step δt. This step is smaller than the mean collision time.

2) The space domain is divided into cells with sides δx, δy, which should be smaller than the local mean free path.

3) The gas molecules are simulated in the channel using a stochastic system of N points having position $x_i(t)$, $y_i(t)$, and velocities $\bar{U}_I(t)$.

4) At any given time there are N_m molecules in the mth cell. The basic DSMC algorithm then consists of five stages.

Stage 1: If sampling is needed, update cell-averaged properties to obtain statistical quantities; otherwise skip this stage.

Stage 2: Select particles to make pairs for collision.

Stage 3: Binary collisions in each cell are calculated without moving the particles: translational and internal energy may be redistributed.

Stage 4: New particles are injected into the domain at boundaries.

Stage 5: The particles are moved, and special action is taken when they hit boundaries.

Stages 1–5 are repeated until run time is over.

In our implementation of a DSMC-IP coupled method, a very similar organization is adopted. The core part (d) is illustrated in Fig. 2, in which a new step is added, represented by a dashed box, which is needed for the IP method. We will describe the implementation of the DSMC-IP method using Fig. 2.

1) In Sample Properties DSMC collects sample information from particles as usual. At the same time IP collects sample information from the preserved information. For example, preserved velocities for a given cell are obtained by averaging the preserved velocities of the particles inside this cell; preserved pressure is computed directly from the preserved number density. Suppose there are N particles in the ith cell, then for DSMC the instantaneous velocity for this cell is obtained by averaging the instantaneous microscopic (DSMC) velocities:

$$\boldsymbol{U}_{\text{DSMC},i} = \sum_{j=1}^{N}\left(\frac{\boldsymbol{U}_{\text{DSMC},j}}{N}\right) \tag{12}$$

At the same time the preserved velocity for the ith cell is derived by averaging the preserved macroscopic (IP) velocities:

$$\boldsymbol{U}_{\mathrm{IP},i} = \sum_{j=1}^{N} \left(\frac{\boldsymbol{U}_{\mathrm{IP},j}}{N} \right) \tag{13}$$

The preserved pressure for the cell is directly derived from the preserved number density along with the isothermal assumption:

$$P_{\mathrm{IP},i} = n_{\mathrm{IP},i} mRT \tag{14}$$

2) In Make Pairs (Fig. 2) first we evaluate the local mean free path from the preserved number density in this cell, then divide the cell into subcells with each side no longer than the mean free path. Particles in each subcell are arranged in pairs.

3) In the Collision part DSMC proceeds as usual. For IP when two particles collide, there is no internal energy redistribution, and their preserved translational velocities are divided equally. This is a simple implementation based on the following supposition: when two groups of particles meet, they merge and develop together. This aspect requires further study in the future.

4) For inlet and outlet pressure Boundary Conditions, we adopt the implementation in Ref. 6, i.e., Eqs. (1–3) are used. In our implementation the mean velocity is not obtained by iteration. Instead, the preserved velocity in the cells next to the boundaries are employed. For IP the preserved velocities change according to the surrounding pressure environment. If the pressure is not in balance, then the velocity keeps changing until the pressure is in balance. Particles injected into the cell have the preserved velocities of the cell. This results in a converging inlet or outlet boundary condition.

5) In the Move part when particles cross an inlet or outlet surface, they are removed. The acceleration is only applied to the preserved velocities of the particles and not to the DSMC velocities. The preserved velocity is changed in each time step by an amount

$$\delta \boldsymbol{u} = \boldsymbol{a} \delta t \tag{15}$$

To maintain stability, a relaxation is usually adopted for the acceleration:

$$\boldsymbol{a}_i = \omega \boldsymbol{a}_{i-1} + (1 - \omega) \boldsymbol{F}_{\mathrm{net}} / m \tag{16}$$

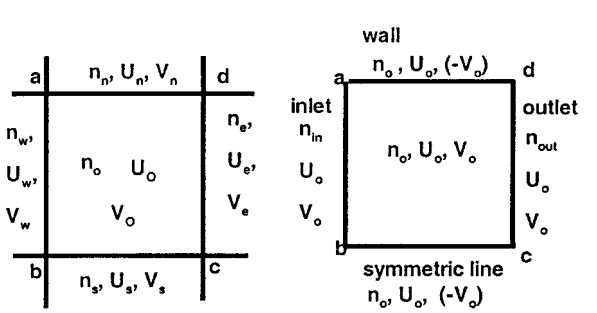
where subscript i represents the ith time step. In this study a relaxation factor of $\omega = 0.5$ is adopted.

Initially, the time step δt for the IP method is a little smaller than mean collision time to maintain stability. When the flow is steady, the time step can be increased. When a particle hits a wall, the preserved velocities are set to zero. This is the IP implementation of diffuse reflection.

6) The additional stage, Preserved Information Update, is a critical component for the IP method.

In this part, first the pressure in all cells is computed directly from the preserved number density. Figure 3 shows the neighboring quantities' relation to the center cell's information. If there are neighboring cells on both sides of a cell edge, the pressure acting on this edge is obtained by averaging the pressures on both sides of the edge. For example, in Fig. 3 the pressure acting on side ab is

$$P_{ab} = (P_w + P_0)/2 = (n_w + n_0) mRT/2 \tag{17}$$



a) All neighbor elements are cells **b) Some neighbor elements are boundaries**

Fig. 3 Preserved information relationship between neighboring cells.

If this is not the case, then the edge must be a special boundary, and we must specify the pressure outside the corresponding side. If the special boundary is a wall or symmetry line, then the pressure on this cell side is the same as the pressure in the cell; if the cell sides are inlet or outlet boundaries, then the pressure is the given pressure at the inlet or outlet to satisfy the pressure boundary condition.

In the second step the net force acting on the cell is computed by summing the product of the pressure along the cell side and the cell side's length, over all cell sides. Then by computing the preserved mass in the cell, the acceleration $\boldsymbol{a}$ is obtained in preparation for the movement in the next time step.

In the third step the preserved number density is updated for every cell in the computation domain at every time step according to mass conservation:

$$\frac{\partial n}{\partial t} + \nabla \cdot (n \boldsymbol{U}) = 0 \tag{18}$$

written in finite volume form:

$$\delta n = -\delta t \int_S (n \boldsymbol{U} \cdot \boldsymbol{e}_n) \frac{\mathrm{d}S}{\delta v} = -\frac{\delta t}{\delta v} \sum_{i=1}^{4} \frac{(n_0 \boldsymbol{U}_0 + n_i \boldsymbol{U}_i)}{2} \cdot \boldsymbol{e}_{n,i} \, \mathrm{d}l \tag{19}$$

where n represents number density, δv is the volume of this control volume, and $\boldsymbol{e}_n$ represents the unit normal vector. The integral is made along cell sides. Subscript 0 represents quantities in a given cell, and subscript i represents the quantities in a neighboring cell. A structured rectangular cell system is adopted. So, when the right-hand-side term is computed, the number density can be updated by the quantity δn.

In the fourth step the preserved velocities in the cell are updated by directly averaging the particles' preserved velocities contained in this cell. Temperature is assumed to be fixed and is not updated.

There is a significant difference in our implementation of pressure boundary conditions in comparison to the former DSMC studies presented in Refs. 6 and 7. Pressure is directly used instead of flux iteration. From the very beginning the desired pressure acting on the boundary is set and never will be changed. We use the preserved velocity instead of the instantaneous velocity from the DSMC method in Eq. (1). We will see that the preserved velocity provides much cleaner results than those obtained from the DSMC method. We avoid adopting Eq. (4), which will result in large numerical or physical fluctuations.

Analysis of the Memory and Running Cost of the IP Method

To couple the DSMC method with the IP method requires increases in memory and run time. In our implementation we use the DSMC code MONACO[12] as a base for the DSMC-IP simulation. Now let us analyze the additional memory and runtime costs for the IP method.

Memory Increase

In the DSMC method each particle has the following information: $x, y, z, u, v, w, E_{\mathrm{rot}}, E_{\mathrm{vib}}$. For the DSMC-IP coupled method three preserved macroscopic velocities are added to each particle: U, V, W. Also for the IP method an array of nine elements is added in each cell to preserve cell information: $U, V, W, P, n, T, \boldsymbol{a}$; another array of four elements is added to record macroscopic sampling results of U, V, W, n in each cell; and at cells near a boundary, an array of five elements is added to record wall quantities from the preserved macroscopic information.

For DSMC the memory cost is

$$\sum_{i=1}^{N\,\mathrm{cell}} [C_1 + 8 \times N_{\mathrm{obj}}(i) + C_2 \times N_b(i)]$$

$$= C_1 \times N\mathrm{cell} + 8 \times N_{\mathrm{obj}} + C_2 \times N_b \tag{20}$$

where Ncell is the total number of cells in the computation domain; C_1 is the number of variables in the DSMC method to maintain

cell information, for example, cell identification number, cell point coordinates; C_2 is the number of variables in the DSMC method to maintain boundary-type information; N_{obj} is the total number of particles; and N_b is the total number of boundary faces.

For the DSMC-IP coupled method the memory cost is

$$(C_1 + 9 + 4) \times N \text{ cell} + (8 + 3) \times N_{\text{obj}} + (C_2 + 5) \times N_b$$

For a structured rectangular cell system suppose we divide the sides into $M \times N$ points. Then the ratio of the number of boundary elements to the number of cell-side elements is $\mathcal{O}[2 \times (M + N)/(M \times N)]$, and further, suppose in each cell we have N_{ave} particles then the cost ratio should be

$$\frac{(\text{DSMC} + \text{IP})}{\text{DSMC}} = \frac{(C_1 + 13) + 11 \times N_{\text{ave}} + (C_2 + 5)(N + M)/(NM)}{C_1 + 8 \times N_{\text{ave}} + C_2(N + M)/(NM)} \quad (21)$$

Therefore, the memory increase is asymptotically 37.5%.

Run-Time Increase

Run-time analysis is far more complex, and detailed discussion would be too lengthy. A brief answer for this question is that in almost every part of the steps shown in Fig. 2 the IP method requires an extra cost of $\mathcal{O}(N_{\text{particles}})$, where $N_{\text{particles}}$ is the total number of particles in the simulation.

Cost in Reaching Steady State

We would like to know the status of the flow when it is approaching steady state in order to reduce the total computational cost. In a CFD scheme residual concepts are employed to monitor convergence. However, because of the statistical scatter problem, we cannot use the same approach for the DSMC method. The traditional approach for DSMC is to monitor the total number of particles in the computation domain. When the total number does not change significantly, we begin to sample data.

One direct benefit from the IP method is that it provides an indicator of the approach to steady state because the IP method can greatly reduce the scatter problem. In this study we introduce the averaged number density n_{ave} and averaged streamwise velocity U_{ave}:

$$n_{\text{ave}} = \sum_{i=1}^{N\text{ cell}} \left(\frac{n_{i,\text{ip}}}{N \text{ cell}/n_{\text{in}}} \right) \quad (22)$$

$$U_{\text{ave}} = \sum_{i=1}^{N\text{ cell}} \left(\frac{U_{i,\text{ip}}}{N \text{ cell}} \right) \quad (23)$$

where N cell is the total number of cells in the computational domain, $n_{i,\text{ip}}$ and $U_{i,\text{ip}}$ are the number density and streamwise velocity in the ith cell in the IP method correspondingly, n_{in} is the inflow number density with which we normalize the number density indicator. These two factors globally indicate the mass and momentum development status in the simulation.

Of course, when these two indicators become steady, it does not necessarily mean that the flowfield is steady: it is only globally steady. Perhaps many more time steps are needed to approach a locally steady state everywhere in the domain. Therefore, for safety, in our study several times more steps are taken to make sure that a steady state has been reached.

Comments

To understand the role of the IP method, we can consider that there are three levels of numerical simulation: CFD, based on the Eulerian continuum field point of view, totally macroscopic; DSMC, based on the Lagrangian particle point of view, totally microscopic; IP, containing collective properties of a group of molecules so that it is submacroscopic and partially microscopic. It is a type of particle method for movement and collision, but it is updated using a finite volume method, with the aid of Eulerian field quantities. The DSMC-IP coupled method is a technique involving the combination of different levels.

The DSMC-IP coupled method has two meanings. One is that the IP method provides the mean velocity at the inlet or outlet for the pressure boundary conditions required by DSMC. Another is that there is only one particle system but two kinds of quantities that develop simultaneously.

The IP and DSMC methods are essentially identical in terms of the sensitivity of the computed results to the choice of numerical parameters. The usual DSMC rules apply: 1) the time step must be less than the mean time between collisions; 2) there must be a sufficient number of particles in each cell to accurately simulate the collision rate (usually we say there must be at least 20 particles per cell); and 3) the cell size should be small enough to resolve flow gradients adequately. As mentioned earlier, the gradients in the x directions for these microchannel flows are about 1000 times smaller than those in the y direction, and this allows use of relatively large cell dimensions in the x direction.

Results

Conditions

Two cases are considered. First, case A, a very short microchannel, is computed to test the IP method and the boundary condition implementation. Case B corresponds to the experimental conditions of Ref. 2. The computational conditions are listed in Table 1. Figure 4 shows the computational geometry for case A. The grid cell size in the y direction is taken to be less than the outlet mean free path as this is the direction of greatest gradients in flow properties. The cell size in the x direction is then obtained by considering the ratio of the gradient length scales in the x and y directions. For example, in case B the gradient length scale for u velocity in the y direction is 1000 times smaller than that for pressure in the x direction. This aspect allows use of cell sizes in the x direction that are significantly larger than a local mean free path.

We employed diffusive wall boundary condition and the variable hard sphere model (Ref. 5) for collisions. A rectangular grid system of 80×50 cells is employed, with 2.5×10^6 particles, and 20,000 time steps of sampling after 30,000 time steps of development.

Other Modeling Techniques

Four types of results are presented. The DSMC-IP implementation provides two results: one from DSMC and one from the

Table 1 Computation conditions

Case	A	B
Gas	O_2	N_2
Channel length L, μm	15	3000
Channel height H, μm	0.53	1.2
Inlet pressure	2.5×10^5 Pa	20 psig
Outlet pressure	1.0×10^5 Pa	0 psig
λ_{out} ($\times 10^{-8}$m)	5.65	6.99
Mean collision time ($\times 10^{-10}$ s)	1.71	1.98
Time step ($\times 10^{-10}$ s)	1.1	1.9
T_{wall}, K	300	300
T_{in}, K	300	300
T_{out}, K	300	300
Cell number	80×50	250×20
Viscosity index ω	0.70	0.70
TMAC coefficient σ	0.85	0.85

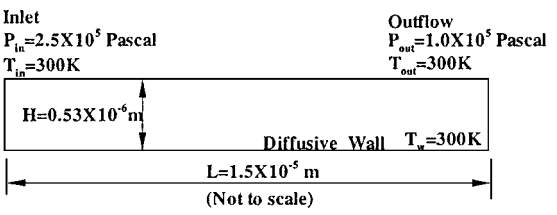


Fig. 4 Microchannel geometry for case A.

preserved information by IP. The other results consist of numerical solutions of the Navier–Stokes equations (N-NS) and analytical solutions of the Navier–Stokes equations (A-NS).

Because the flow inside the microchannel is slow and the length scale is small, the Reynolds number is very small, and so we can neglect the convective term in the Navier–Stokes equations. Further, combined with slip boundary conditions

$$U|_{\text{wall}} = \frac{2-\sigma_m}{\sigma_m}\lambda\frac{\partial U}{\partial y}\bigg|_{\text{wall}} \tag{24}$$

where y is along the lateral direction and σ_m is the tangential momentum accommodation coefficient (TMAC), whose value is set from experimental results, we can obtain the A-NS solution. The main analytical results are listed next. More detailed information is available in Refs. 4, 9, and 10. The accuracy of the A-NS results is first order:

$$P(x) = \sqrt{\left(6Kn_o\frac{2-\sigma}{\sigma} + P_i)^2 - \frac{x}{L}\left(\left(P_i^2-1\right) + 12Kn_o\frac{2-\sigma}{\sigma}(P_i-1)\right)\right.}$$

$$\left. -6Kn_o\frac{2-\sigma}{\sigma}\right. \tag{25}$$

$$U(x) = \frac{1}{2\mu}\frac{\mathrm{d}p}{\mathrm{d}x}\left(y^2 - \frac{H^2}{4} - H^2Kn\frac{2-\sigma}{\sigma}\right) \tag{26}$$

$$\dot{m} = \frac{H^3 w P_o^2}{24\mu LRT}\left(P_i^2 - 1 + 12\frac{2-\sigma}{\sigma}Kn_o(P_i-1)\right) \tag{27}$$

where x is the streamwise distance from the inlet; y is the lateral distance from the centerline; $P(x), U(x), \dot{m}$ are nondimensional pressure distribution, dimensional streamwise velocity and mass flux rate; Kn is Knudsen number; subscript o means values are based on the outlet value; P_i represents the ratio of inlet pressure to outlet pressure; σ_m is set to 0.85 to fit experimental results; channel width w is set to unit value 1.0; ω is the viscosity temperature exponent.[5]

The N-NS results are computed using a suite of CFD methods developed at the University of Minnesota.[13–15] These methods use a conventional finite volume formulation with implicit time integration to reduce the cost of obtaining steady-state results. The implicit methods have been designed for use on parallel computer architectures. The primary difficulty for the CFD technique is associated with the low flow speeds and dominance of the viscous effects. Most existing CFD methods are designed for flows with a relatively high Reynolds number where convective processes dominate. Also, because of the low flow speeds, the compressible Navier–Stokes equations are poorly conditioned. As a result, the convergence properties of conventional CFD methods are very poor for these flows. In this study the microchannel simulation uses a method that includes the effects of all viscous terms in the implicit operator. This allows very large time steps to be taken during convergence to a steady state. In this case a Courant–Friedrichs–Lewy number of 10^9 was used. This indicates that had an explicit time integration method or a conventional implicit method been used the simulations would require about 4000 iterations to converge to a steady state; this is compared to less than 100 for flows that are not as viscous dominated.

Results and Comparison

Figures 5a and 5b show the pressure contours and the centerline pressure distributions. Pressures are normalized by the outlet pressure. The four kinds of results overlap one another and are not easily discernible. Nonlinear distributions are clearly displayed along the centerline, and this behavior is also observed in experimental results. These comparisons indicate that the IP method provides a result close to the other three methods.

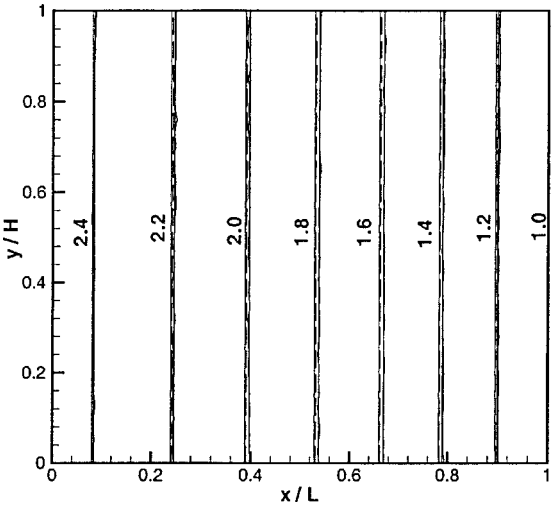


Fig. 5a Comparison of pressure contours: ——, N-NS; – – –, DSMC; – –, A-NS; and · · · , IP.

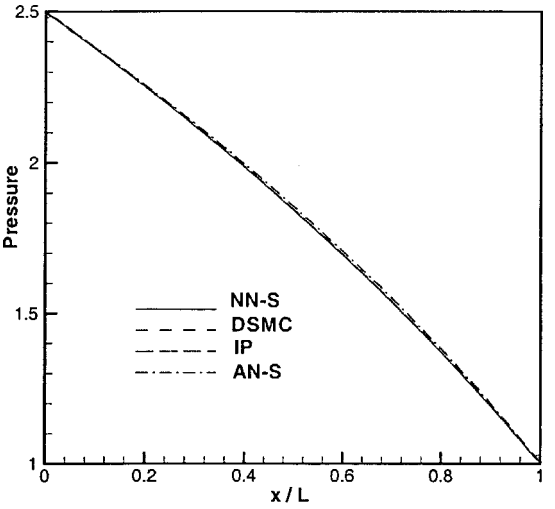


Fig. 5b Comparison of centerline pressure distribution.

Figures 6a–6d show x-velocity contours and distributions along the centerline at $x = L/3$ and at $x = 2L/3$. We can see that N-NS and the DSMC method give consistent results. The IP results are also close to these methods. We can see from Figs. 6c and 6d that IP also shows better symmetry than the DSMC method.

Figures 7a and 7b show the y-velocity distributions at $x = L/3$ and $2/3L$ by DSMC, IP, and N-NS. Previously, it was believed[8] that the DSMC method cannot reproduce the near-wall region velocities in the y direction because these values are so small that the scatter dominates the results. However, we can see that the IP method successfully reproduces the symmetry property in this test case and that the values are very near to the N-NS results. IP captures the small values successfully not only in trend but also in magnitude. The discontinuity in the N-NS solutions is caused by the finite volume method adopted in the computation, which cannot give exact values at the walls. The velocity values are too small for the DSMC method to provide accurate results with a reasonable number of samples.

Figure 8 shows the centerline translational temperature distribution from DSMC and N-NS. The purpose of this plot is to check how well the isothermal assumption holds. The system error with this assumption for the IP method is $(300 - 297)/300 = 1.0\%$ at most. If we compute flow under experimental conditions, the isothermal assumption will be even better because the longer the channel, for the same inlet to outlet pressure ratio, the smaller the changes in the streamwise velocity, which will result in smaller

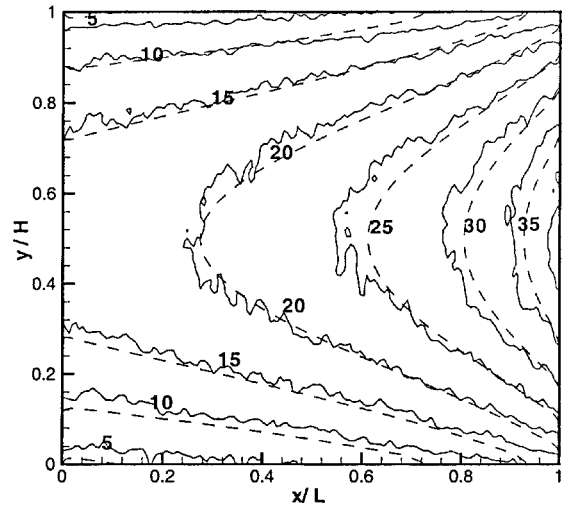


Fig. 6a Comparison of x-velocity contours: ---, N-NS, and ——, DSMC.

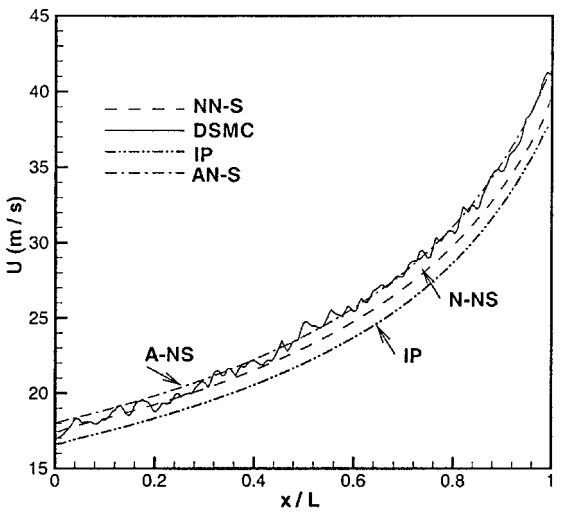


Fig. 6b Comparison of centerline x-velocity distributions.

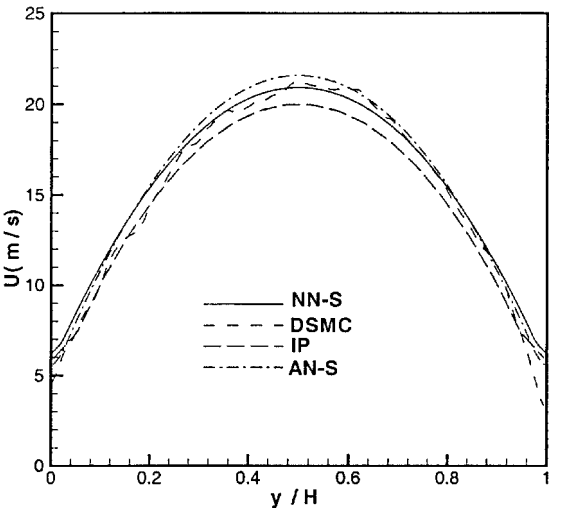


Fig. 6c Comparison of x-velocity distributions at cross section $x = L/3$.

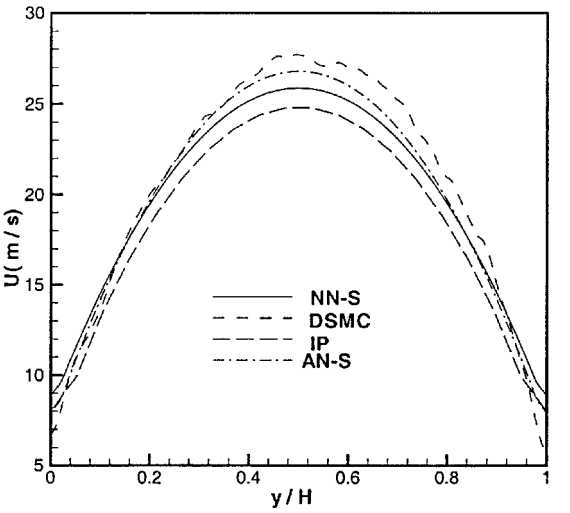


Fig. 6d Comparison of x-velocity distributions at cross section $x = 2L/3$.

temperature changes. One possible reason accounting for the temperature difference between the results in Fig. 8 is the different implementations of the boundary conditions in the CFD and DSMC methods.

Figure 9 shows the upper-wall slip velocities (the velocity in the cell nearest the wall). The IP result fits much better to the N-NS solution than the DSMC method. However, this does not mean that DSMC is wrong. The present wall reflection model for the IP method is too simple and needs further consideration.

Figure 10 shows the wall shear stress from DSMC and IP. From this plot we can see that the wall shear stress predicted by IP is similar to the result from DSMC. However there is clearly a large amount of scatter in the DSMC results, whereas the IP method shows almost no scatter.

It is difficult to say which method will most accurately predict the mass flux rate. Because the mass flux rate formula of the analytical solutions A-NS [Eq. (27)] is supported by experimental results, it is convenient to use it as a standard result for comparison. For DSMC, IP, N-NS, and A-NS solutions, we compute the numerical integration

$$\dot{m} = \int_0^H \rho u \, dy \tag{28}$$

at sections $x = L/3$ and $2/3L$, where the symmetry is maintained. The results are shown in Table 2. From this table we can see that all four methods give good agreement, indicating the success of our DSMC-IP coupled implementation.

Now let us concentrate on the comparison of IP and DSMC methods for scatter and implementation of pressure boundary conditions. Table 3 gives the mean x velocity U and standard deviation σ_u of the particle velocity at the inlet ($x = 0$) and outlet cells ($x = L$) from the DSMC method. These data are evaluated in one time step, like a snapshot. Here,

$$U = \sum_{i=1}^{N} \frac{u_i}{N}$$

$$\sigma_u^2 = \sum_{i=1}^{N} \frac{(u_i - U)^2}{N}$$

where N is the number of particles in this cell and u_i is the velocity of a specific particle: similar definitions are used for V and σ_v. From this table we can see clearly that the problem in DSMC is that the standard deviations are two or more orders of magnitude higher than the signals. We will need 90,000 particles in this cell in one time step to suppress the noise, rather than the 1000 particles used in the current implementation. Iteration methods, such as those proposed in Refs. 6 and 7, would therefore appear to be inappropriate.

Table 4 gives the counterpart of Table 3 from the IP method, from which we can see that the noise is decreased by two orders of

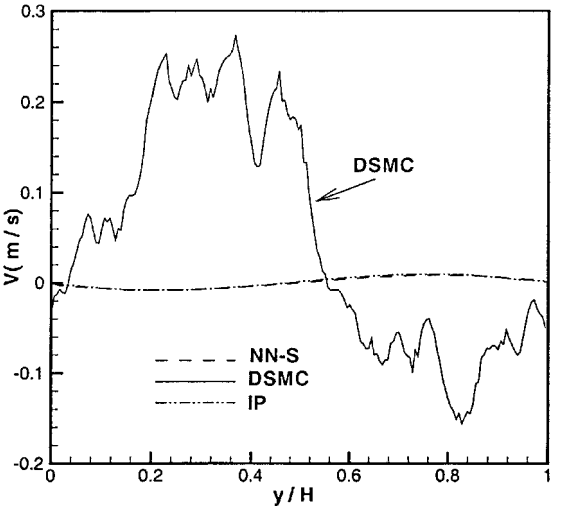


Fig. 7a Comparison of y-velocity distributions at cross section $x = L/3$ from DSMC, IP, and N-NS.

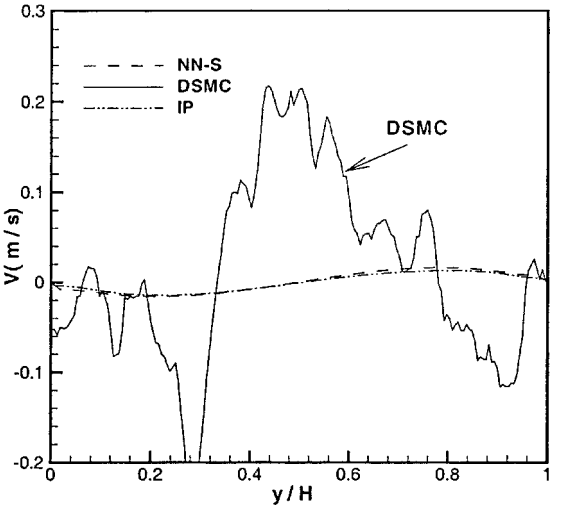


Fig. 7b Comparison of y-velocity distributions at cross section $x = \frac{2}{3}L$ from DSMC, IP, and N-NS.

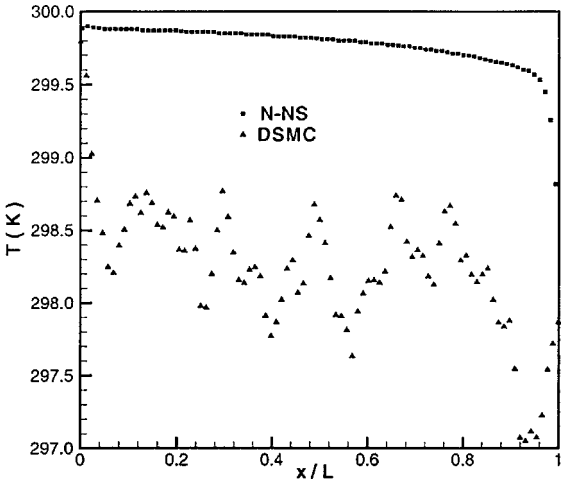


Fig. 8 Comparison of centerline translational temperature distributions.

Table 2 Comparison of mass flux rate

Method	$\dot{m}$, kg/sm^2	$\Delta\dot{m}$, %	$\dot{m}$, kg/sm^2	$\Delta\dot{m}$,%
Eq. (27)	2.1691×10^{-5}	——	2.1691×10^{-5}	——
A-NS	2.1931×10^{-5}	+1.1	2.1938×10^{-5}	+0.1
DSMC	2.1661×10^{-5}	−0.1	2.2193×10^{-5}	+2.3
IP	2.0701×10^{-5}	−4.5	2.0791×10^{-5}	−4.2
N-NS	2.1575×10^{-5}	−0.47	2.1415×10^{-5}	−1.3

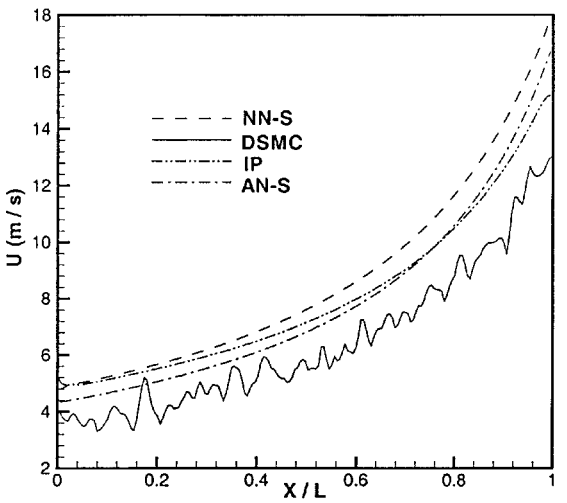


Fig. 9 Comparison of lower-wall slip velocities from four methods.

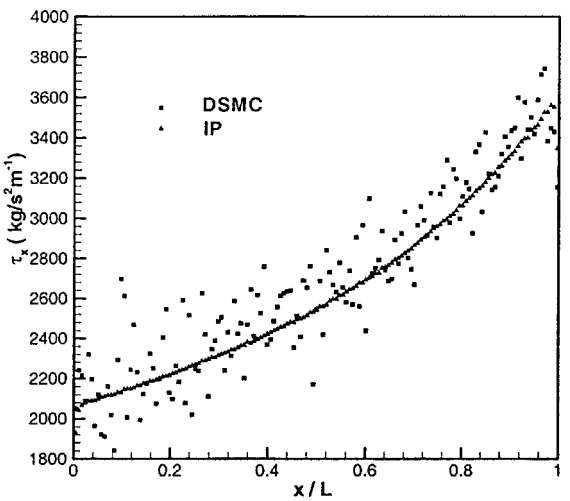


Fig. 10 Comparison of wall shear stress τ_x from DSMC and IP methods.

magnitude in the x direction and four orders of magnitude in the y direction. We can also see from Tables 3 and 4 that, at the inlet and outlet, the IP method gives much better symmetry in the streamwise direction distributions than DSMC.

Figure 11 gives the particle velocity phase by IP and DSMC at an inlet cell in the position $(x, y) = (0, H/2)$ at a time after steady flow is established. We can see that particles cluster together in a small region in the IP method, whereas in the DSMC method they spread equally away along the x and y directions. Results at other locations are similar. This picture demonstrates the advantage of the IP method for low-speed flow.

Figures 12a and 12b show the developing history of average streamwise velocity in an inlet cell with the position $(x, y) = (0, H/2)$ from IP and DSMC. To save time, the initial flowfields of velocity and number density distributions are given by the A-NS results. From Figs. 12 we see the developing history of the streamwise velocity: first it decreases very rapidly, then there is a period of large

Table 3 Mean velocity and standard deviation in different cells from the DSMC method

x/L	y/H	U, m/s	V, m/s	σ_u, m/s	σ_v, m/s
0	0	7.2267	−3.5078	279.1926	281.5711
0	1/4	17.6531	−2.9091	276.0072	270.1264
0	1/2	24.9815	−1.2742	278.0757	276.4445
0	3/4	18.8677	−0.7286	282.6543	277.2055
0	1	2.7456	8.5289	277.2512	280.5454
1	0	5.2654	21.5183	277.8687	272.4186
1	1/4	21.1762	−6.1319	282.0066	283.1123
1	1/2	24.8911	2.6713	276.9211	272.4414
1	3/4	56.5116	2.7724	277.7005	270.8946
1	1	7.3997	8.5340	280.3568	288.3394

Table 4 Mean velocity and standard deviation in different cells from the IP method

x/L	y/H	U, m/s	V, m/s	σ_u, m/s	σ_v, m/s
0	0	4.1266	0.0479	2.9227	0.0429
0	1/4	13.3922	0.0866	1.4685	0.0123
0	1/2	15.6439	0.0657	0.5101	0.0089
0	3/4	13.3479	0.0395	1.4718	0.0091
0	1	4.2234	0.0077	2.8818	0.0066
1	0	13.7281	0.0138	11.7387	0.0188
1	1/4	33.5875	0.0303	6.7617	0.0251
1	1/2	37.7043	0.0226	4.2261	0.0217
1	3/4	33.6191	0.0106	6.6079	0.0303
1	1	13.1091	−0.0653	11.8531	0.0231

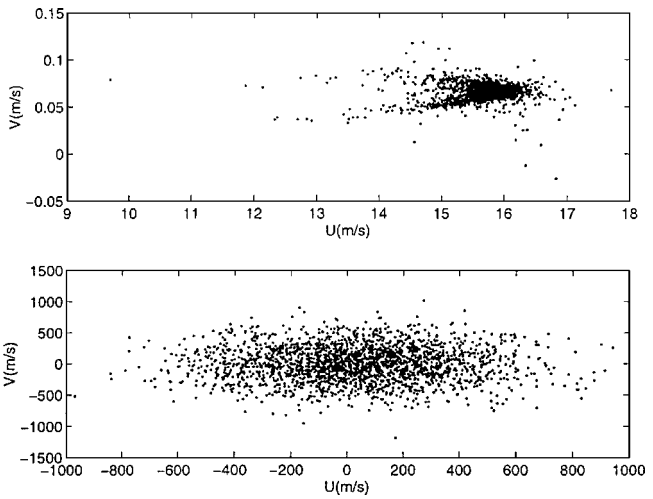


Fig. 11 Comparison of velocity phases in an inlet cell at $(0, H/2)$ from IP (top) and DSMC (bottom).

fluctuations, then the fluctuations become smaller and smaller. If we average the velocity over a longer time span then use it as the mean velocity for the inlet or outlet, the scatter will be further suppressed, leading to better results. Figure 12a also indicates that evaluation of the cost of establishing the steady process by the IP method is possible. From Fig. 12b we can see that implementations of pressure boundary conditions based on a mean velocity evaluation from DSMC are inappropriate.

Figures 13a and 13b show the approach to steady state in the IP method from different initial flowfields. We select three initial flowfields: one is from A-NS; the second initial condition is just a low density, zero velocity flowfield; and the last one is a developed flowfield from DSMC but with a perturbation pulse at location $x = L/3$. Constant lines are included in each plot to indicate the steady value. From these figures we can see, for both mass and momentum, that the initial field of A-NS solutions develops fastest. Although

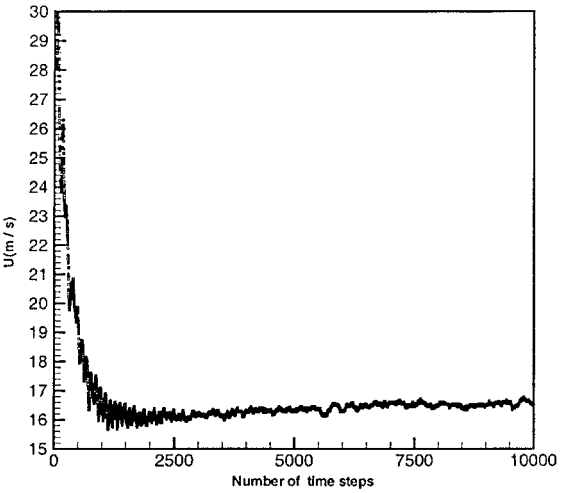


Fig. 12a Evolution history of x velocity in an inlet cell at $(0, H/2)$ from IP.

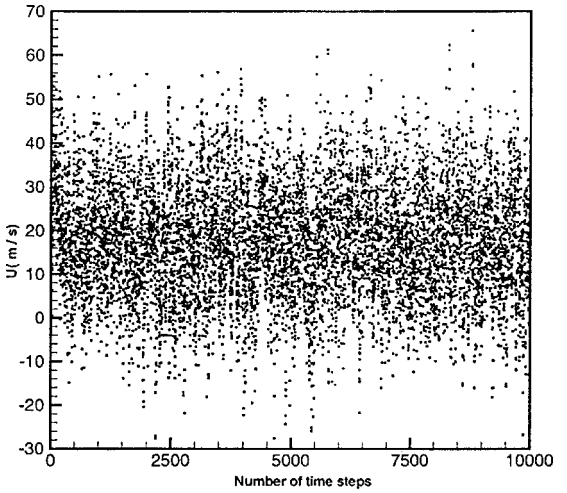


Fig. 12b Evolution history of x velocity in an inlet cell at $(0, H/2)$ from DSMC.

the perturbation is small, it needs a longer time to develop; and the last one, i.e., the void initial flowfield, develops slowest. We can conclude from these two plots that using a good initial solution is very helpful in reducing the cost of reaching steady state with the IP method.

Figure 14 shows changes in particle velocity occurring in the IP method caused by the net pressure force and by the collisions inside the cell at location $x = L/2$, $y = H/2$. We can see that these two mechanisms change a particle's velocity almost equally. This is perhaps explained by the fact that the Reynolds number is 1 or so, indicating that the inertial force, i.e., the net pressure force acting on this cell, and the internal viscous force, i.e., caused by the particle collision and movement inside the cell, are equally important.

Case B is for the experimental conditions of Ref. 2. It is the shortest microchannel investigated experimentally. Figure 15 shows the centerline pressure distribution, from which we can see that there is almost no difference between the three kinds of results. (There are no N-NS results available for this case.) This is because the degree of rarefaction is not high enough to make the Navier–Stokes equations become invalid.

Figure 16 shows the centerline velocity distribution from IP and A-NS. There is too much statistical scatter in the DSMC result. The IP method successfully provides small velocity distribution.

The significance of these results is that this is the first time that it has been possible to simulate microchannel flow and obtain meaningful results under experimental conditions with a particle method.

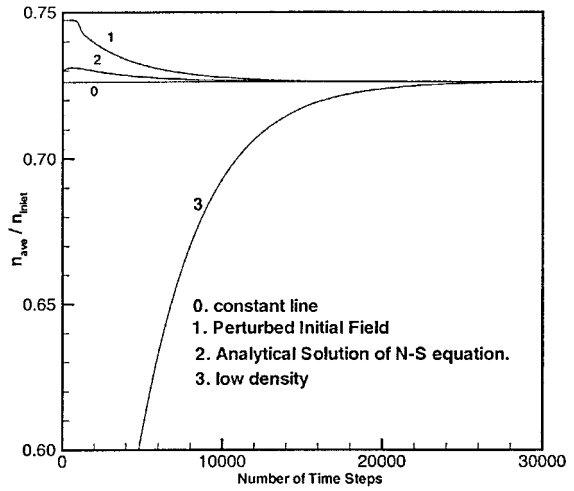


Fig. 13a Comparison of average number density developing history from different initial fields from the IP method.

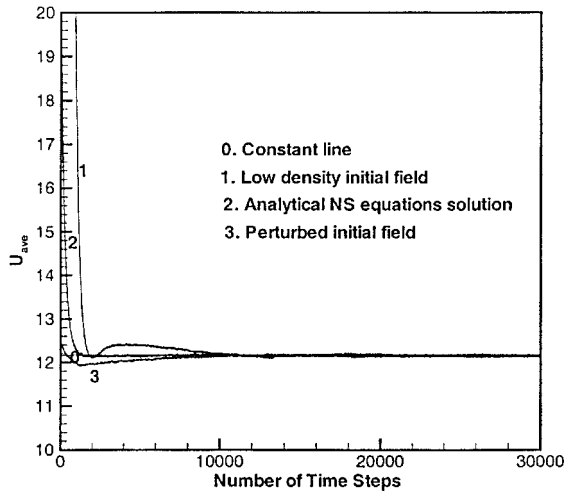


Fig. 13b Comparison of average x velocity developing history from different initial fields from the IP method.

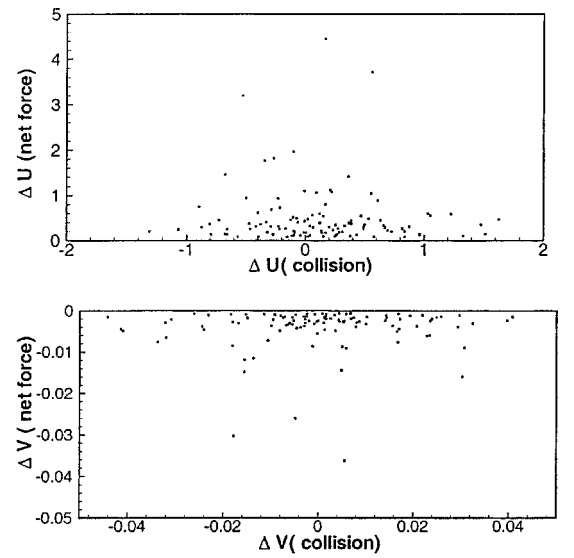


Fig. 14 Particles velocity change in the IP method by net pressure force and collision within one time step inside a cell at location ($L/2$, $H/2$).

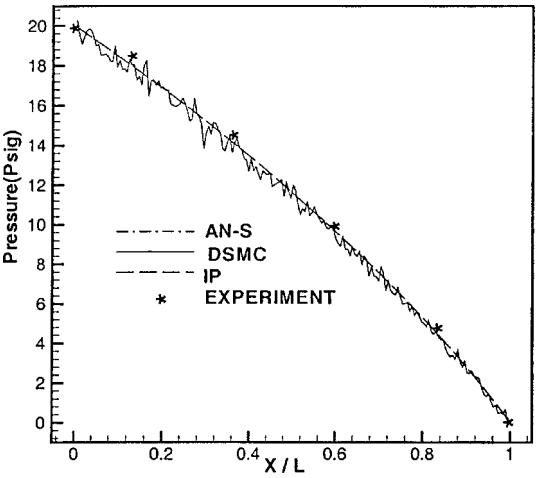


Fig. 15 Centerline pressure distribution for case B.

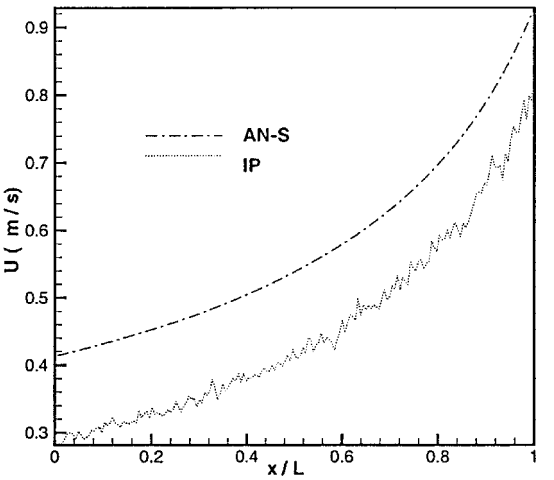


Fig. 16 Centerline velocity distribution for case B.

Conclusion

From the preceding results we can clearly see that the DSMC-IP method successfully reproduces the flowfield. All four methods considered provide similar results for pressure distribution. The DSMC-IP coupled method gave mass flux rates very close to the analytical result of Eq. (27) and solutions for velocity distributions that were in good agreement with numerical and analytical solutions of the Navier–Stokes equations, no matter how small the velocity. Previously, it was believed impossible for DSMC to capture such slow velocity values.[8] The DSMC-IP coupled method successfully extends the DSMC method to small velocities. In this study we found that IP can reduce the sampling cost by a factor of 10^2–10^4.

In summary, in this study we obtained answers to several problems.

1) Reduction of sampling cost after a flow becomes steady: The preserved information of the IP method has significantly smaller scatter than DSMC properties, and therefore sampling is only required over a small number of times steps.

2) Implementation of the pressure boundary conditions: The IP method can quickly provide a much cleaner velocity locally and at the current time, i.e., there is no need to enlarge the sampling area to neighboring cells or to use multitime step averaging. This positive property can minimize the undesirable effects associated with the previous DSMC pressure boundary implementations of Refs. 6 and 7.

3) Reduction of the cost in approaching steady state: Use of a good initial state is critical. From this study we can see that the relaxation time needed is not of the order $\mathcal{O}(L/U)$, where L is the channel length and U is the inlet velocity. It is even smaller than

$\mathcal{O}(L/a)$, where a is the sonic speed. A good initial state will require only a fast local relaxation.

Although the IP method has been successfully extended to two dimensions, it is still a new method under development. There are several aspects to be improved. For other applications it may be necessary to remove the isothermal assumption. This requires preservation of the translational temperature and can also require consideration of internal energy distribution. The present IP models for collision and wall reflection are very simple and require further study. Despite the need for further development, the conclusion is made that the DSMC-IP method in its present form has significantly advanced our ability to apply particle methods to MEMS flows.

Acknowledgment

The authors gratefully acknowledge support for this work from the Air Force Office of Scientific Research through MURI Grant F49620-98-1-0433 to the University of Minnesota.

References

[1]Ho, C. M., and Tai, Y. C., "Micro-Electro-Mechanical Systems (MEMS) and Fluid Flows," *Annual Review of Fluid Mechanics*, Vol. 30, 1998, pp. 579–612.

[2]Pong, K. C., Ho, C. M., Liu, J. Q., and Tai, Y. C. "Nonlinear Pressure Distribution in Uniform Microchannels," *Application of Microfabrication to Fluid Mechanics*, FDRL Vol. 197, American Society of Mechanical Engineers, 1994, pp. 51–56.

[3]Shih, J. C., Ho, C. M., Liu, J. Q., and Tai, Y. C., "Monatomic and Polyatomic Gas Flow Through Uniform Microchannels," *Microelectromechanical Systems (MEMS)*, DSC-Vol. 59, American Society of Mechanical Engineers, 1996.

[4]Arkilic, E. B., "Measurement of the Mass Flow and Tangential Momentum Accommodation Coefficient in Silicon Micromachined Channels," Ph.D. Dissertation, Dept. of Aeronautics and Astronautics, FDRL TR97-1, Massachusetts Inst. of Technology, Cambridge, MA, Jan. 1997.

[5]Bird, G. A., *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Clarendon, Oxford, 1994.

[6]Ikegewa, M., and Kobayashi, J., "Development of a Rarefield Gas Flow Simulation Using the Direct-Simulation Monte Carlo Method," *JSME International Journal*, Series 2, Vol. 33, No. 3, 1990, pp. 463–467.

[7]Nance, R. P., Hash, D., and Hassan, H. A., "Role of Boundary Conditions in Monte Carlo Simulation of MEMS Devices," *Journal of Thermophysics and Heat Transfer*, Vol. 12, No. 3, 1998, pp. 447–449.

[8]Oh, C. K., Oran, E. S., and Sinkovvits, R. S., "Computations of High-Speed, High Knudsen Number Microchannel Flows," *Journal of Thermophysics and Heat Transfer*, Vol. 11, No. 4, 1997, pp. 497–505.

[9]Arkilic, E. B., Schmidt, M. A., and Breuer, K. S., "Gaseous Slip Flow in Long Microchannels," *Journal of MicroElectroMechanical Systems*, Vol. 6, No. 2, 1997, pp. 167–178.

[10]Piekos, E. S., and Breuer, K. S., " DSMC Modeling of Micromechanical Devices," *Journal of Fluids Engineering*, Vol. 118, 1996, pp. 464–469.

[11]Fan, J., and Shen, C., "Statistical Simulation of Low-Speed Unidirectional Flows in Transitional Regime," *Rarefied Gas Dynamics*, Vol. 2, edited by R. Brun, Cepadues Editions, Toulouse, France, 1999, p. 245.

[12]Dietrich, S., and Boyd, I. D., "Scalar and Parallel Optimized Implementation of the Direct Simulation Monte Carlo Method," *Journal of Computational Physics*, Vol. 126, 1996, pp. 328–341.

[13]Candler, G. V., Wright, M. J., and McDonald, J. D., "A Data Parallel LU Relaxation Method for Reacting Flows," *AIAA Journal*, Vol. 32, No. 12, 1994, pp. 2380–2386.

[14]Olejniczak, D. J., and Candler, G. V., "A Data-Parallel LU Relaxation Method for DNS of Compressible Flows," *Proceedings of the First Air Force Office of Scientific Research International Conference on DNS and LES*, Paper N-03, 1997.

[15]Wright, M., Candler, G. V., and Prampolini, M., "Data Parallel Lower-Upper Relaxation Method for Navier-Stokes Equations," *AIAA Journal*, Vol. 34, No. 7, 1996, pp. 1371–1377.