AIAA JOURNAL
Vol. 40, No. 1, January 2002

Nonlinear Filtering for Low-Velocity Gaseous Microflows

Carolyn R. Kaplan* and Elaine S. Oran†
U.S. Naval Research Laboratory, Washington, D.C. 20375

Gaseous flows in microfluidic devices are often characterized by relatively high Knudsen numbers. For such flows, the continuum approximation is not valid, and direct simulation Monte Carlo (DSMC) may be used to find an appropriate solution. For low-velocity flows, where the fluid velocity is much smaller than the mean molecular velocity, large statistical fluctuations in the solution mean that the features of the flow may be obscured by noise in the solution. The use of a high-order, nonlinear monotone convection algorithm, flux-corrected transport (FCT), as a filter to extract the solution from the noisy DSMC calculation is described. The diffusion, antidiffusion, and flux-limiting properties of FCT are discussed in terms of their filtering properties. The effects of filtering with FCT are demonstrated for a series of test problems, including a square wave with superimposed random noise, and low- and high-velocity and low- and high-Knudsen-number microchannel flows. It is shown that FCT filtering removes high-frequency statistical fluctuations and can extract a solution from a noisy DSMC calculation when the flow velocity is much less than the molecular thermal velocity and when the ratio of real to simulated particles is relatively high. Although the application described has been limited to microflows, this filter has significant potential for a wide variety of DSMC problems used in physical regimes where statistical noise must be eliminated. Because it is a postprocessing operation and does not affect the DSMC calculation as it is running, filtering can be applied to any DSMC solution. This includes calculations with complex geometries, the presence of many interacting species with chemical reactions, varied and complicated boundary conditions, and time-dependent solutions.

Nomenclature

f	=	flux
H	=	channel height
i	=	spatial location (subscript)
Kn	=	Knudsen number
Kn_o	=	Knudsen number at outlet
k	=	Boltzmann constant
L	=	characteristic length, channel length
m	=	molecular mass
P	=	dimensionless pressure distribution
$P_{i/o}$	=	inlet to outlet pressure ratio
T	=	fluid temperature
t	=	time
t_{fluid}	=	fluid transit time in channel
$U(x)$; V_x	=	streamwise velocity
V_{inflow}	=	inflow velocity
V_{th}	=	most probable thermal velocity
v; V_{flow}	=	fluid velocity
x	=	streamwise distance along channel
y	=	vertical distance from centerline of channel
λ	=	mean free path
μ	=	dynamic viscosity, antidiffusion coefficient
ν	=	antidiffusion coefficient
ρ	=	fluid density, generic variable
σ	=	accommodation coefficient

Superscripts

d	=	diffused solution
da	=	diffused and antidiffused solution
dal	=	diffused, antidiffused, and flux-limited solution
n	=	final updated solution
o	=	original solution
td	=	transported and diffused solution
tdal	=	transported, diffused, antidiffused, and flux-limited solution

Received 30 December 2000; revision received 30 April 2001; accepted for publication 30 June 2001. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0001-1452/02 $10.00 in correspondence with the CCC.

*Research Chemical Engineer, Laboratory for Computational Physics and Fluid Dynamics, Code 6410; ckaplan@lcp.nrl.navy.mil. Senior Member AIAA.

†Senior Scientist, Laboratory for Computational Physics and Fluid Dynamics, Code 6400. Fellow AIAA.

Introduction

MANY proposed and existing microelectromechanical systems have at least one component that involves low-velocity gas flows in micrometer-sized components. Flows in such small chambers may be characterized by relatively large Knudsen numbers, $Kn \equiv \lambda/L$, where λ is the mean free path and L is a characteristic length. When the Knudsen number is large enough, for example, $Kn > 0.1$ or even less, the flow is nonequilibrium, and many of the usual assumptions about continuum representations begin to break down. As the Knudsen number increases, the continuum approximation loses its validity and eventually is simply a wrong approximation. For a gas in a microdevice at 1 atm and 298 K, the Knudsen number is on the order of 0.15–0.20. Flows in the range $Kn \sim 0.1$–10, for which continuum assumptions break down, are in a transitional flow regime between Euler flows and free molecular flow (see, for example Ref. 1).

One of the most common and robust methods for computing the properties of such transitional flows is direct simulation Monte Carlo (DSMC), a statistical method that uses a number of simulated particles to represent typical particles in the flow. The method has been applied to a wide variety of flows, such as flows around reentry vehicles, chemical vapor deposition, and plasma processing. It has been used to describe flows in extremely complex three-dimensional geometries and with complicated chemical reaction mechanisms. Many of the details of how to include the complexities of reactive flows and surface effects are active areas of research.

There is, however, a serious problem with trying to use DSMC to simulate flows in microdevices, in which there are relatively low flow velocities and large number densities.[2] For example, consider a flow at 1 atm and 298 K, for which the mean collision time is $\sim 10^{-10}$ s. The mean particle transit time for a relatively slow flow of 10 cm/s through a microchannel of length 1000 μm is on the order of 10^{-2} s. Because the DSMC time step must be less than the mean collision time, it would take an enormous number of timesteps (10^8) for the particles to make one complete pass through the entire computational domain.[3] For the low-velocity flow problem, the small time step is devastating because it typically requires times on the order of several fluid transit times through the computational domain for the system to reach steady state.

An additional problem occurs when the flow velocity V_{flow} is much smaller than the most probable thermal velocity $V_{\text{th}} = \sqrt{(2kT/m)}$, where k is the Boltzmann constant, T is the fluid temperature, and m is molcular mass.[2] For a light gas, such as helium, at 1 atm and 298 K, $V_{\text{th}} \simeq 100{,}000$ cm/s. If $V_{\text{flow}} = 200$ cm/s, the statistical noise is over a factor of 10^2 greater than the solution, and the features of the flow are easily lost in the noise. In principle, the correct result should emerge as the ensemble or time average converges because the statistical fluctuations decrease with the number of samples. However, an enormous number of samples are required to reduce the noise in such a low-velocity flow.

Because of these problems, DSMC calculations of gaseous microflows have generally been carried out only for high-velocity flows,[4–8] whereas the low-velocity calculations have been made with continuum methods incorporating slip boundary conditions.[9–13] More recently, methods have been developed to use DSMC to simulate low-velocity microflows. One of the most recent methods is the Information Preservation[14] technique, in which each simulated molecule is characterized by two velocities: the ordinary thermal velocity that is used to compute the molecular trajectory and an information velocity that is aimed at recording the collective velocity of the enormous number of real molecules represented by a simulated molecule. After collision between two simulated molecules, the particles are assigned the same postcollision information velocity based on a prescribed collision model. The macroscopic velocity and shear stress are obtained from the information velocities of the simulated molecules through time or ensemble averaging and, therefore, are much less noisy than those based on molecular thermal velocities in standard DSMC methods. Another modified DSMC method for low-velocity flows is based on the concept of splitting the random velocity of a gas molecule, as well as the stress and temperature of the gas flow, into two independent parts: One part is determined through a macroscopic physical model, and the other is obtained by the DSMC algorithm. With this method,[15] the thermal kinetic energy of the molecule does not have any effect on the collision computation.

In this paper, we consider another method to reduce the problem of statistical noise: applying a nonlinear filter to the noisy DSMC solution. There have been extemely sophisticated filtering methods devised to reduce noise from acoustic and electromagnetic signals. Many of these take advantage of some known property of the object to be detected or analyzed. Here we examine the use of a nonlinear filter, flux-corrected transport, which is routinely used in fluid dynamics problems to eliminate high-frequency noise and ensure monotonicity in convection algorithms.

Test Problem

Principles of DSMC

DSMC is a particle simulation method based on kinetic theory. The fundamental idea is to track a large number of statistically representative particles. The particles' motion and interactions are then used to modify their positions, velocities, or chemical reactions. Conservation of mass, momentum, and energy is enforced. The primary approximation of DSMC is to uncouple the molecular motions and the intermolecular collisions over small time intervals. Particle motions are modeled deterministically, whereas the collisions are treated statistically. Modeling the collisions is always a three-dimensional calculation, even when the dimensions of the computational grid are reduced to account for symmetries in physical space.

The core of the DSMC algorithm consists of four primary processes: move the particles, index and cross reference the particles, simulate collisions, and sample the flowfield. These procedures are uncoupled during each time step. Of primary importance is the selection of a time step that is smaller than the local mean collision time and a cell size that is smaller than the local mean free path. The first process, moving simulated molecules, enforces the boundary conditions and samples macroscopic properties along solid surfaces. Modeling molecule–surface interactions requires applying the conservation laws to individual molecules instead of using the velocity distribution function. The second process involves indexing and tracking the particles. Accurate and fast indexing and tracking are key to practical DSMC applications for large-scale processing. The next step is simulating collisions between randomly selected pairs of molecules within each cell. This step is a probabilistic process. There are several different collisional modeling techniques that have been used in DSMC codes over the years. In this work, we use the no-time-counter technique in conjunction with the subcell technique.[1] The subcell method calculates local collision rates based on the individual cells, but restricts possible collision pairs to subcells. This procedure improves accuracy by ensuring that collisions occur only between nearest neighbors. The final process is sampling the macroscopic flow properties. The other steps of the DSMC procedure do not depend on the sampling process.

Flowfield sampling occurs at specified intervals. At these times, each particle's molecular mass and velocity components are cumulatively summed within each computational cell. These summed velocity quantities are then used to calculate the macroscopic properties in that cell. When the number of samples is large, more molecules are included in the cumulative sum in each cell, which helps to reduce the statistical noise in the solution.

Because DSMC is an explicit, time-marching technique, it can be used for either time-dependent or steady-state computations. For steady-state computations, each computation proceeds until a steady flow is established, and then the results are time averaged and sampled to determine the final state. In the case of unsteady flows, the results are sampled at specified intervals. The criterion for the DSMC time step is analogous to the Courant limit on the sound speed imposed on explicit methods in fluid dynamics algorithms.

Statistical scatter is an inherent part of DSMC calculations. Because each simulated molecule represents a large number of actual molecules, the collision computation is stochastic. There is a tradeoff here: Increasing the number of simulated molecules does reduce the statistical error but increases the computational work. Numerical accuracy in DSMC depends on the grid resolution chosen as well as the number of particles per computational cell. As discussed in more detail hereafter, we have tested both.

Another kind of input data is the collision cross section, which is used to model the effects of chemical reaction. The models of collision cross section may range from the simplest hard sphere (HS) model to complex models for inelastic collisions that account for vibrational and rotational nonequilibrium. The HS model is sometimes adequate for idealized monatomic gas computations but not for real gases. The variable hard sphere (VHS) model[16] corrects for the primary deficiency in the HS model, namely, an inaccurate representation of the total collision cross section, while retaining simplicity of implementation. In the VHS model, the effective collision cross section decreases as the relative translational energy increases. There are more sophisticated collision models that have been reviewed extensively in the literature.[1]

Slow Flow in a Channel

The test problem is a two-dimensional, $1 \times 12\ \mu\text{m}^2$ channel filled with helium at 1 atm and 298 K. Helium enters the channel at velocities in the range from 200 to 80,000 cm/s. (In this range, we confine ourselves to systems for which we do not expect to see the formation of shocks.) For helium at 298 K, $V_{\text{th}} = 110{,}000$ cm/s. We expect the statistical fluctuations to affect the solution significantly when the flow velocities are 10,000 cm/s or less. The pressures at the inflow and outflow boundaries are set to 1 atm.

For all of the calculations shown here, the grid consisted of 150×30 cells and approximately 100,000 simulated VHS molecules. The channel boundaries are modeled as diffuse walls, so that the velocity of a molecule after reflection from the wall is randomly determined from the appropriate thermal distribution set by the wall temperature (298 K).

First, consider the steady-state, converged, and statistically averaged results of the relatively high-velocity $V_{\text{inflow}} = 60{,}000$ cm/s case shown in Fig. 1. Although the inflow velocity at the boundary is fixed for the course of the computation, the mean x velocity near the inlet is reduced to approximately 10,000 cm/s as the pressure builds up to over 2 atm. The velocity contours show boundary layers forming at the inlet and growing downstream on the upper and lower walls. The temperature throughout the channel remains almost

constant, except for a slight increase at the inlet and slight decrease at the outlet. For this case (Fig. 1), numerical accuracy was tested by doubling the grid resolution and the number of simulated molecules, that is, 300×60 cells and 200,000 particles. These resolution tests are not shown because they resulted in no appreciable differences in the solution and in our conclusions. This is a relatively uneventful flow. At higher inflow velocities, the flow structure is more complex and the ranges (minimum to maximum) of variables in the computational domain are larger. If the velocity is high enough, the flow structure also changes substantially as an assortment of shock waves form and interact (see, for example, Refs. 3 and 7).

Figure 2 shows computed x velocities, pressures, temperatures, and densities for a series of calculations in which the inflow velocities were systematically decreased from 60,000 cm/s to 200 cm/s. The computed range of values of these variables (shown just above each part of Fig. 2) contracts as the flow velocity decreases; that is, the flows become more uniform throughout the microchannel as the inflow velocity decreases. The physical time to which each calculation is carried out is shown in Table 1. For the highest inflow velocity shown, $V_{\text{inflow}} = 60{,}000$ cm/s, it was possible to obtain smooth, converged solutions in a few hours. Even after a few days of computation, however, the solutions for $V_{\text{inflow}} = 1000$ cm/s were not smooth, and the situation becomes impossible for $V_{\text{inflow}} = 200$ cm/s. In particular, the values of V_x for this case are extremely noisy and it is essentially impossible to see any trend unless you already know the solution. Drawing a contour map (such as that shown in Fig. 1a for the $V_{\text{inflow}} = 60{,}000$ cm/s solution) would give no information for this $V_{\text{inflow}} = 200$ cm/s case. For low-velocity cases, varying the grid

Table 1 Relevant times for results shown in Fig. 2

V_{inflow}	Physical time of results, μs	Fluid transit time along centerline t_{fluid}, μs	Computer time, approximate h
60,000	20	0.08	4
10,000	100	0.6	20
5,000	150	1.2	30
1,000	200	8.0	40
200	1,000	15.0	200

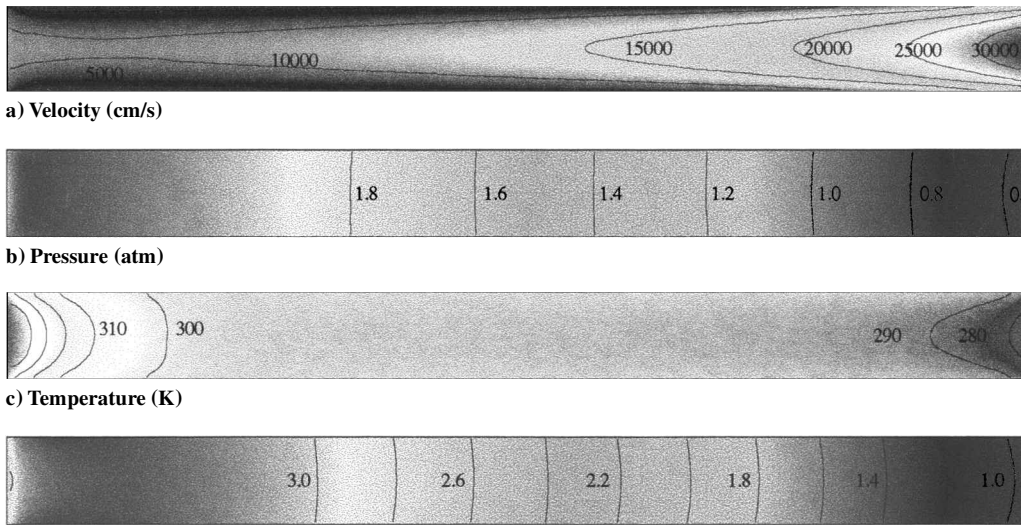

Fig. 1 Selected macroscopic quantities for high-velocity case, $V_{\text{inflow}} = 60{,}000$ cm/s.

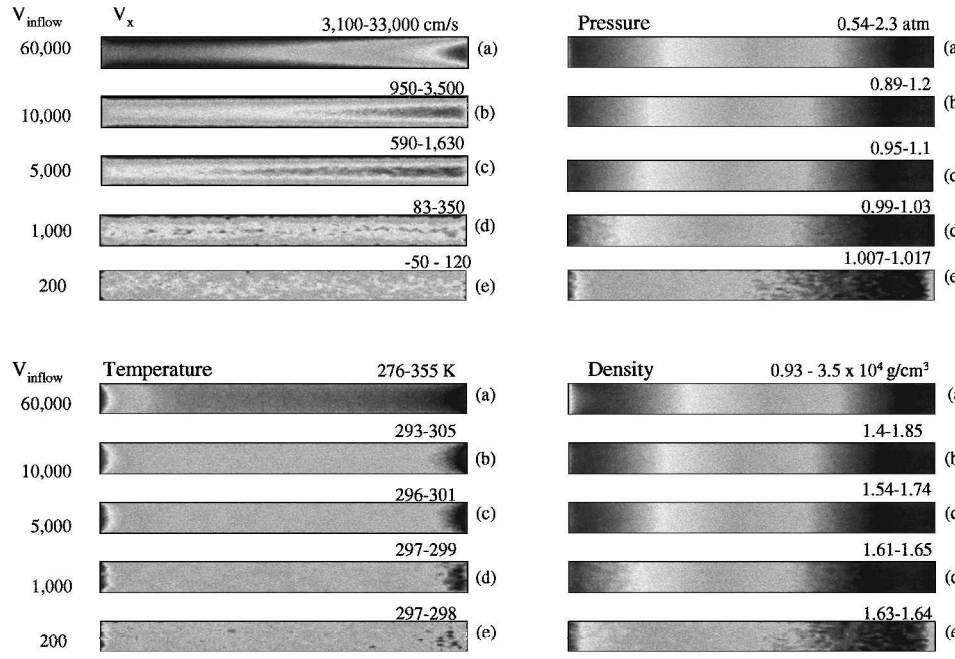

Fig. 2 Macroscopic quantities for a range of inflow velocities; min/max ranges for each variable are shown above each image.

resolution and the number of particles per cell within any reasonable range does not affect the solution because the problems asociated with statistical noise are overwhelming.

The essence of the problem, then, is to find a general, reliable way to extract the low-velocity solution. Ideally, this method would allow for complex geometries, the presence of many interacting species, varied and complicated boundary conditions, and time dependence of the solution.

Nonlinear Filtering Properties of Flux-Corrected Transport

How do we extract an unknown solution from what appears to be a field of noise? In the most general case, we do not know what the signal really is, only that there is or might be one in the solution. We consider filtering two sets of data: 1) noisy data taken from a very early time of the high-velocity $V_{\text{inflow}} = 60{,}000$ cm/s case and 2) the $V_{\text{inflow}} = 200$ cm/s case taken from Fig. 2e. In the first case, $V_{\text{inflow}} = 60{,}000$ cm/s, we know the final, converged solution (as shown in Fig. 2a), and we can compare that to the results of filtering the early-time, noisy calculation. In the second case, we do not know the final solution, although we have some idea of its general properties.

Figure 3 shows the results of putting the noisy high-velocity early-time $V_{\text{inflow}} = 60{,}000$ cm/s case and the low-velocity $V_{\text{inflow}} = 200$ cm/s case through a standard, diffusively smoothing algorithm for 1, 50, 100, and 1000 passes. This filter can be described as a simple arithmetic averaging of each point with its eight surrounding points. A disquieting feature of these results is that they continue to change with time, which raises the question of when to stop filtering. This could vary as the problem changes and, perhaps, for a complicated flow, according to the location in the flow. We would like to use a filter for which the flow converged in some orderly way to a solution. Comparison of the diffusively filtered result for the high-velocity case ($V_{\text{inflow}} = 60{,}000$ cm/s, 1000 passes) with the converged DSMC solution (in Fig. 2a) indicates that the diffusive filter does not provide the correct solution. Therefore, we cannot have any confidence that the diffusive filter is correctly removing statistical noise from the low-velocity $V_{\text{inflow}} = 200$ cm/s case either, as shown here. This will be addressed later when the nonlinear filter is applied to these two cases.

This issue is reminiscent of the problem occurring in early, diffusive algorithms for solving continuity equations. Such algorithms formed the bases for solving the coupled continuity equations, with source terms, that comprise the Navier–Stokes equations. For example, one early algorithm is a first-order donor-cell method that results in a continuously more diffuse solution as time advances. It was later shown that linear higher-order algorithms, which were less diffusive, were more dispersive and ultimately unstable. The solution to this came with nonlinear monotone algorithms. The first of these was flux-corrected transport (FCT),[17,18] which introduced the concepts of positivity, monotonicity, and flux limiting into the language of solutions of continuity equations. FCT was followed shortly by a series of algorithms that also ensure positivity, monotonicity, and limit fluxes, and later by the suite of total variation diminishing algorithms. Further details of the FCT algorithm are given in Ref. 19. Descriptions of general classes of these algorithms are given in Refs. 19 and 20.

FCT was originally designed to solve a continuity equation for a variable ρ:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho v = 0 \tag{1}$$

by a series of steps that globally conserve ρ, do not accentuate any existing extrema, and do not introduce any new maxima or minima into the solution. Specifically, the values $\{\rho_i^o\}$, where i indicates the spatial location of ρ, are advanced in time to $\{\rho_i^n\}$. This is achieved by a multistep process:

1) The initial values ρ^o are transported and diffused, giving ρ^{td}. The diffusion here ensures positivity and stability.

2) To remove excess diffusion, antidiffusion fluxes $f^{\text{da}}(\rho^{\text{td}})$ are defined. The antidiffusion stage, however, can introduce negative values or nonphysical overshoots in the solution.

3) Because of step 2, the antidiffused fluxes are limited to ensure positivity and stability, before they are applied to ρ^{td} to find ρ^n.

This ensures that no new unphysical maxima or minima are added to the solution. Schematically, then, we could write

$$\rho_i^o \longrightarrow \rho_i^{\text{td}} \longrightarrow \rho_i^{\text{tdal}} \equiv \rho_i^n \tag{2}$$

There have been many mathematical and physics treatises on how and why FCT and other flux-limiting algorithms work. Here we present one interpretation of how it works as a filter and, in doing this, explain some of the basic ideas of the algorithm.

When FCT is used as a filter, there is no time advancement. Thus, step 1 is only diffusion, and $\rho^{\text{td}} = \rho^d$. Now define the flux of ρ, $f_{i+1/2}(\rho)$, at the location halfway between ρ_i and its neighbor ρ_{i+1}, by

$$f^d_{i+\frac{1}{2}}(\rho^o) = \nu_{i+\frac{1}{2}}\left(\rho^o_{i+1} - \rho^o_i\right) \tag{3}$$

where $\{\nu_{i+1/2}\}$ are diffusion coefficients. Then

$$\rho^d_i = \rho^o_i + f^d_{i+\frac{1}{2}}(\rho^o) - f^d_{i-\frac{1}{2}}(\rho^o) \tag{4}$$

which is an intermediate value of ρ. Next, raw antidiffused fluxes are defined:

$$f^{\text{da}}_{i+\frac{1}{2}}(\rho^d) = \mu_{i+\frac{1}{2}}\left(\rho^d_{i+1} - \rho^d_i\right) \tag{5}$$

where $\{\mu_{i+1/2}\}$ are antidiffusion coefficients. If these fluxes were used directly to evaluate ρ^n, new maxima and minima could be introduced. Therefore, step 3 limits f^{da} by defining

$$f^{\text{dal}}_{i+\frac{1}{2}} = S \cdot \max\Big\{0, \min\Big[S \cdot \left(\rho^d_{i+2} - \rho^d_{i+1}\right), \left|f^{\text{da}}_{i+\frac{1}{2}}\right|, S \cdot \left(\rho^d_i - \rho^d_{i-1}\right)\Big]\Big\} \tag{6}$$

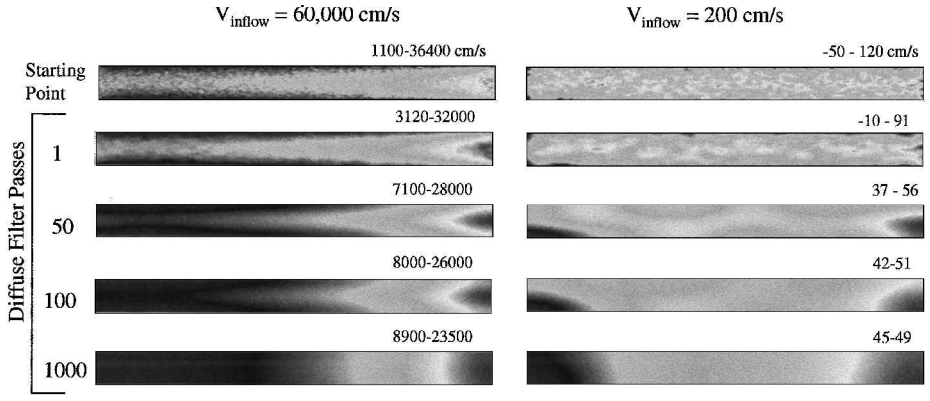

Fig. 3 Diffusive filtering results for V_{inflow} = 60,000 and 200 cm/s cases. Min/max ranges for each variable are shown above each image. Starting point for filtering operation corresponds to when DSMC calculation is taken out to a physical time of 0.12 μs (slightly larger than the fluid transit time, 0.08 μs) for the V_{inflow} = 60,000 cm/s case, and out to 1000 μs (much larger than the fluid transit time, 15 μs) for the V_{inflow} = 200 cm/s case.

where $|S| = 1$ and sign $S = \text{sign}\,(\rho^d_{i+1} - \rho^d_i)$. Then

$$\rho^n_i \equiv \rho^{\text{dal}}_i = \rho^d_i - f^{\text{dal}}_{i+\frac{1}{2}}(\rho^d) + f^{\text{dal}}_{i-\frac{1}{2}}(\rho^d) \tag{7}$$

The effect of this process is first to smooth local peaks in the data, a process that alters the local value of ρ and the value of its neighbors. Then, in a limited antidiffusion step, the neighbors keep value closer to their original values, but the peak remains smoothed. The basic algorithm can vary in the choices of the diffusion and antidiffusion coefficients and in the exact way in which the fluxes are limited. In the filtering tests described hereafter, $\nu = \mu = \frac{1}{6}$, and the fluxes are limited as shown in Eq. (6). Examining the consequences of varying these three choices could lead to improved filtering properties.

The practical result of these steps is a high-frequency filter with some particularly useful properties, as shown in Fig. 4. Here a square wave, propagating at zero velocity [$v = 0$ in Eq. (1)], is passed through the FCT algorithm using a uniform grid, $\Delta x = 1$, as shown in Fig. 4a. The result is that the square wave is unaltered by FCT, for as many passes as considered, up to numerical round off. Now superimpose a random spectrum of high-frequency noise on the square wave, as shown in Fig. 4b. FCT reduces the noise while rigorously conserving ρ. What is most interesting for our problem is that this operation results in a less-noisy solution that stops changing as the number of filter passes increases. An important key to this process is the flux limiting, and it is the flux limiter and the remnants of the antidiffusion that persist when a quantity is convected at zero velocity. The inherent filtering properties of FCT are used in the monotonically integrated large-eddy simulation algorithms for computing turbulent flows.[21,22]

Filtering the Test Problems

High-Speed, High-Knudsen-Number Flow, with Known Solution

For the high-velocity $V_{\text{inflow}} = 60{,}000$ cm/s case, we compare the converged DSMC solution with an unconverged FCT-filtered solution. The filtering was done using the same fixed, evenly spaced grid that was used for the DSMC calculation. One pass through the filter consists of passing the DSMC data once through FCT with $v = 0$ in the x direction and then once in the y direction.

Figure 5 shows the results of filtering a DSMC solution for the high-velocity case, $V_{\text{inflow}} = 60{,}000$ cm/s. The filtering is begun after the DSMC calculation has been carried out to 0.12 μs of physical time, which is a little more than a fluid transit time ($t_{\text{fluid}} = 0.08$ μs) for this case. Figure 5 shows the filtered solution at 1, 50, 500, 1000, 5000, and 10,000 filter passes. First note that passing the solution through many passes of the filter results in a solution that essentially stops changing in time. The final filtered solution shown, at 10,000 passes (Fig. 5g), is not as smooth as the the late-time converged DSMC solution (Fig. 5h), but the two solutions are qualitatively similar. In particular, the boundary layer in the filtered solution is not as well developed as in the converged solution. The filtered solution, however, was based on a DSMC computation carried out to 0.12 μs (and then filtered), but the converged DSMC solution was carried out to 20 μs.

The possibility exists that the filter is picking out the correct solution for the earlier time at which the filtering was started. This seems reasonable because DSMC is an explicit, time-marching method. To begin to investigate this, we began filtering at an even earlier time (0.03 μs) and a later time (0.40 μs), as summarized in Table 2. Figure 6 shows that filtering at the earliest time produces a solution somewhat different from the others. The next case, in which filtering began at 0.12 μs, which is slightly more than a fluid transit time, results in a solution that is similar to the converged DSMC solution. The last case, in which filtering began at 0.40 μs, which is much larger than a fluid transit time, results in a solution that is very similar to the late-time converged DSMC solution. In all three cases, the times at which the filtering began were longer than the sound transit time, which is 0.013 μs. The filtered solution at 0.03 μs is considerably different from the late-time converged DSMC solution, however, it may be the correct solution for that earlier time in the flow.

Figure 7 shows a quantitative comparison of velocity, pressure, and temperature along the centerline longitudinal axis for the converged DSMC and the three filtered solutions discussed earlier. The solution for the late-time converged DSMC solution is practically identical to the filtered solution at 0.40 μs. When the filtering is started earlier, at 0.12 μs, the filtered solution is close to the late-time

Table 2 Relevant information for Figs. 6 and 7, for $V_{\text{inflow}} = 60{,}000$ cm/s

Physical time of results, μs	Computer time, min	Comment
0.03	2	$t < t_{\text{fluid}}$
0.12	8	$t \sim t_{\text{fluid}}$
0.40	25	$t > t_{\text{fluid}}$

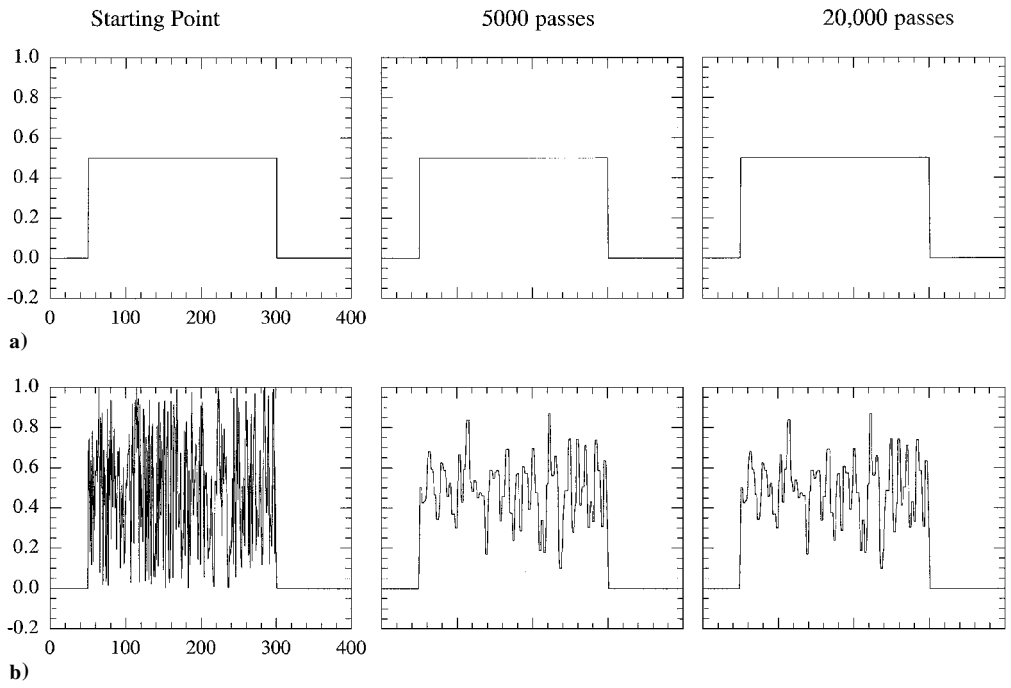

Fig. 4 Plain square wave (panel a) and one that has been superimposed with random noise (panel b) are passed through FCT with $v = 0$; the starting point and the results after 5000 and 20,000 passes through the FCT filter are shown for both cases.

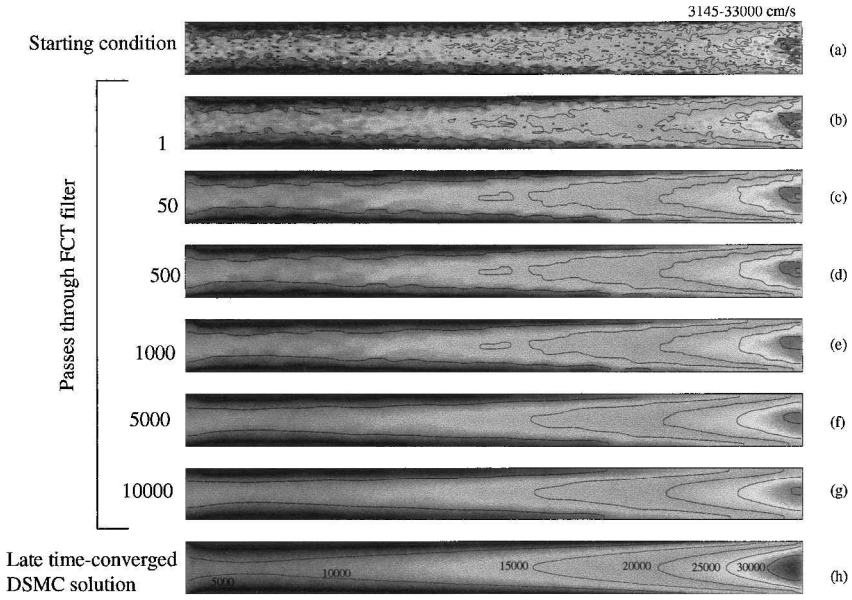

Fig. 5 Passing an early-time solution for $V_{\text{inflow}} = 60{,}000$ cm/s through the FCT filter for indicated number of passes. Starting point for filtering operation corresponds to when the DSMC calculation is taken out to a physical time of 0.12 μs, which is slightly larger than the fluid transit time through the centerline. For comparative purposes, the late-time converged DSMC solution is shown in panel h. The min/max range 3145–33,000 cm/s is the same for all of the images shown.

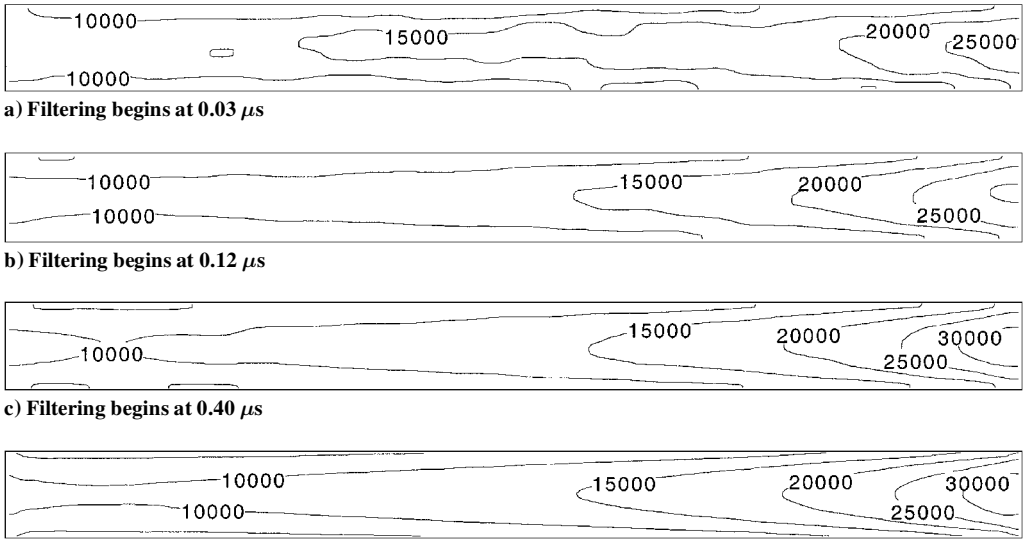

Fig. 6 Axial velocity contours for $V_{\text{inflow}} = 60{,}000$ cm/s: a) the filtering begins at 0.03 μs, which is less than a fluid transit time; b) filtering begins at 0.12 μs, which is slightly more than a fluid transit time; c) filtering begins at 0.40 μs, which is larger than a fluid transit time. For comparative purposes, the late-time converged DSMC solution is shown in panel d.

converged DSMC solution, but not exactly the same. When the filtering begins very early, at 0.03 μs, which is much earlier than a fluid transit time, there is considerable difference compared to the late-time converged DSMC solution.

The computer time required to obtain the late-time converged DSMC solution (Fig. 5h) was approximately 4 h on one processor of an Origin 2000. The time required to obtain the final filtered solution (Fig. 5g; 10,000 passes) is approximately 15 min: That is, it took about 8 min to obtain the noisy early time DSMC calculation (Fig. 5a; starting condition) plus an additional 7 min for FCT filtering (to get from Fig. 5a to Fig. 5g). The length of time required to filter a specific variable adequately depends on how noisy that variable is. For example, V_x is quite noisy and requires more filter passes than a variable such as pressure, which is fairly smooth and only requires a few filter passes.

We have compared the results of the simulations to the analytical solution of the Navier–Stokes equations described by Arkilic.[23] This solution is obtained by neglecting the nonlinear inertial term in the Navier–Stokes equations. Arkilic developed expressions for the velocity and pressure along the centerline longitudinal axis:

$$P(x) = \sqrt{(6\,Kn_o\,[(2-\sigma)/\sigma] + P_{i/o})^2 - (x/L)\left\{\left(P_{i/o}^2 - 1\right) + 12\,Kn_o[(2-\sigma)/\sigma](P_{i/o} - 1)\right\}} - 6\,Kn_o\,[(2-\sigma)/\sigma] \tag{8}$$

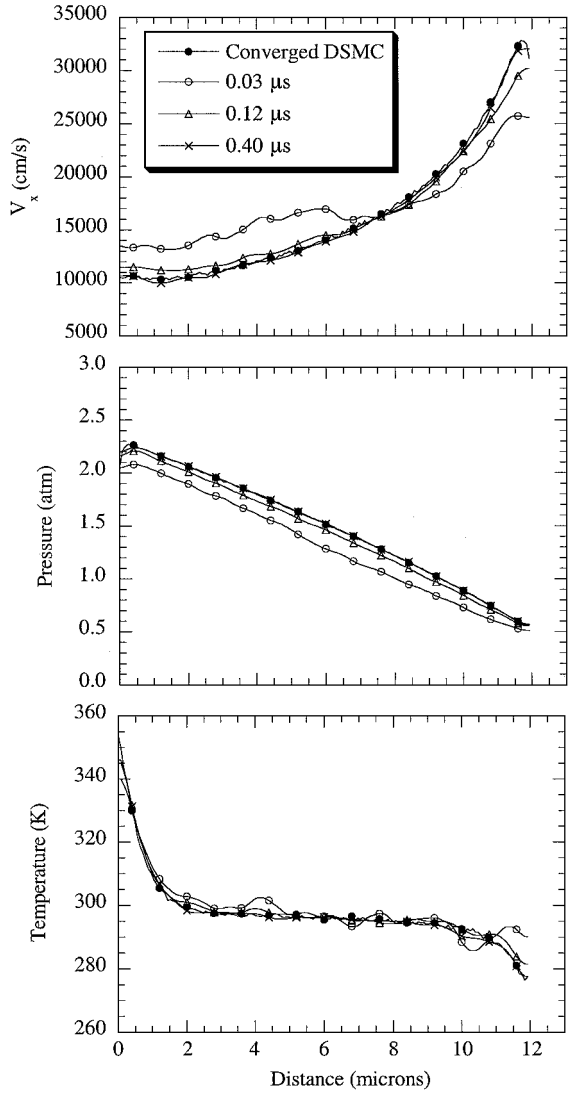

Fig. 7 Axial velocity, pressure, and temperature along the longitudinal centerline for the late time converged DSMC solution (at 20 μs of physical time) and for the three filtered cases shown in Fig. 6.

$$U(x) = \frac{1}{2\mu}\frac{\mathrm{d}P}{\mathrm{d}x}\left(y^2 - \frac{H^2}{4} - H^2\,Kn\,\frac{2-\sigma}{\sigma}\right) \tag{9}$$

where $P(x)$ is the dimensionless pressure distribution, Kn_o is the Knudsen number at the outlet, σ is the accommodation coefficient (assumed to be 0.85), $P_{i/o}$ is the inlet to outlet pressure ratio, x is the streamwise distance from the inlet, L is the channel length, $U(x)$ is the streamwise velocity, μ is the dynamic viscosity, y is the vertical distance from the centerline, and H is the channel height. This solution incorporates a slip boundary condition at the walls:

$$U|_{\text{wall}} = \frac{2-\sigma}{\sigma}\lambda\frac{\partial U}{\partial y}\bigg|_{\text{wall}} \tag{10}$$

The accuracy of the analytical solution is zero order, and it is valid when the Reynolds number is very small. Because we will be dealing with problems for which we simply do not know the exact solution, for example, the low-speed flows, comparisons to analytical solutions, even if they are limited in their ranges of validity, helps give some confidence in the DSMC and filtered DSMC trends that we see.

Figure 8 compares the results of Eqs. (8–10) to the converged solution of the $V_{\text{inflow}} = 60{,}000$ cm/s case. Although this is a relatively high-velocity flow, the Reynolds number for this case is only six, due to the very small length scale; in adddition, this flow is still subsonic because the sound speed is 100,000 cm/s. The DSMC and analytical solutions for the pressure distribution along the centerline both show a nonlinear pressure drop; the differences between the two pressure solutions are within 10% of each other. The velocity profiles are in reasonable agreement, except near the outlet, where the local Knudsen number is highest ($Kn > 0.2$) and where the slip flow conditions assumed in the analytical solution may no longer be valid.

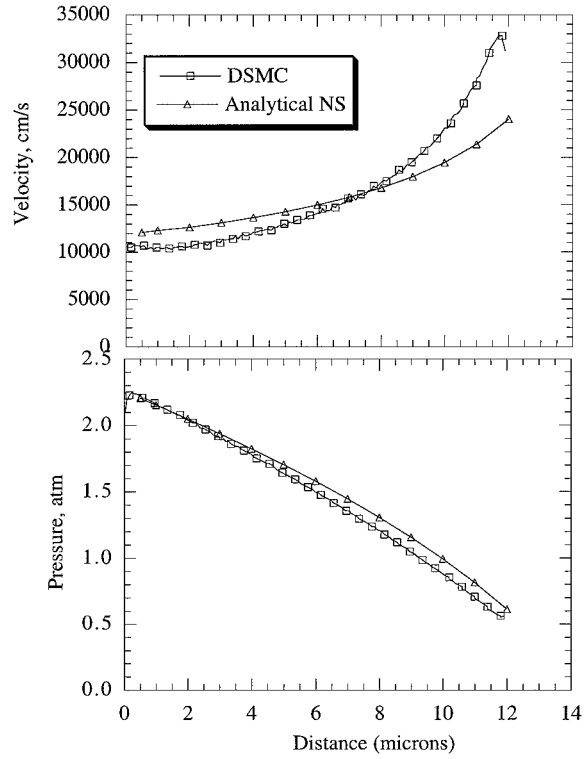

Fig. 8 For the $V_{\text{inflow}} = 60{,}000$ cm/s case, comparison between late-time converged DSMC solution and an analytical solution to the Navier–Stokes equations for axial velocity and pressure along the longitudinal centerline.

Low-Knudsen-Number Flow

Next, we examine FCT filtering of a low-Knudsen-number flow, in which the microchannel contains nitrogen gas at 298 K and 1 atm. For this case, $V_{\text{inflow}} = 10{,}000$ cm/s, which is approximately $\frac{1}{5}V_{\text{th}}$, and $Kn = 0.05$. Figure 9 contains x velocity and pressure plots along the microchannel centerline for an early-time DSMC solution (at 0.8 μs of physical time), a late-time DSMC solution (at 8.0 μs), the early-time DSMC solution that has been FCT filtered, and the analytical solution to the Navier–Stokes equation.[23] As expected, there is considerable noise in the x velocity data for the early-time DSMC calculation at 0.8 μs, although this physical time corresponds to more than the fluid transit time. There is still some noticeable noise in the x velocity data for the late-time DSMC solution, although it corresponds to more than 10 fluid transit times. The results indicate that the filtering of the early-time DSMC solution eliminates much of the statistical noise and is qualitatively and quantitatively similar to the late-time DSMC solution. In addition, the late-time DSMC and the early-time filtered solutions are closer to the analytical Navier–Stokes solution, compared to the higher-Knudsen-number case shown in Fig. 8. This is because the slip flow condition incorporated in the analytical solution is more valid at a lower Knudsen number. Figure 9 shows that the pressure profile is considerably less noisy than that for velocity and also indicates that the DSMC filtered-DSMC, and analytical solutions are very similar for this low-Knudsen-number flow.

The filtered early-time solution took approximately an order of magnitude less CPU time to compute than the late-time DSMC solution. This indicates that FCT filtering is a good approach to eliminating statistical noise for low-Knudsen-number problems. It is typically quite expensive to use DSMC to simulate

low-Knudsen-number flows, because these high-number-density flows require a large ratio of real to simulated particles, which exacerbates the problem of statistical noise. Nonlinear filtering makes it significantly more cost effective to use DSMC to simulate these low-Knudsen-number flows.

Low-Speed, High-Knudsen-Number Flow

The preceding discussion demonstrated the FCT filtering technique for a high-velocity flow because we know the correct DSMC solution for this case. The real benefit of nonlinear filtering is to allow us to use DSMC to simulate a low-velocity flow, where the statistical noise overwhelms the flow solution. Figure 10 shows the $V_{\text{inflow}} = 200$ cm/s case before and after filtering with FCT. The left-hand side shows the DSMC solution after 1000 μs of physical time. Although this time is considerably longer than the fluid transit time for this case, which is 15 μs, the solution (especially the V_x image) is still very noisy. The right-hand side of Fig. 10 shows the result of putting the $V_{\text{inflow}} = 200$ cm/s solution through 15,000 passes of the FCT filter, which required about 5 min of computer time. The result is a smooth solution, although the long-wavelength oscillation shown in V_x is probably unphysical.

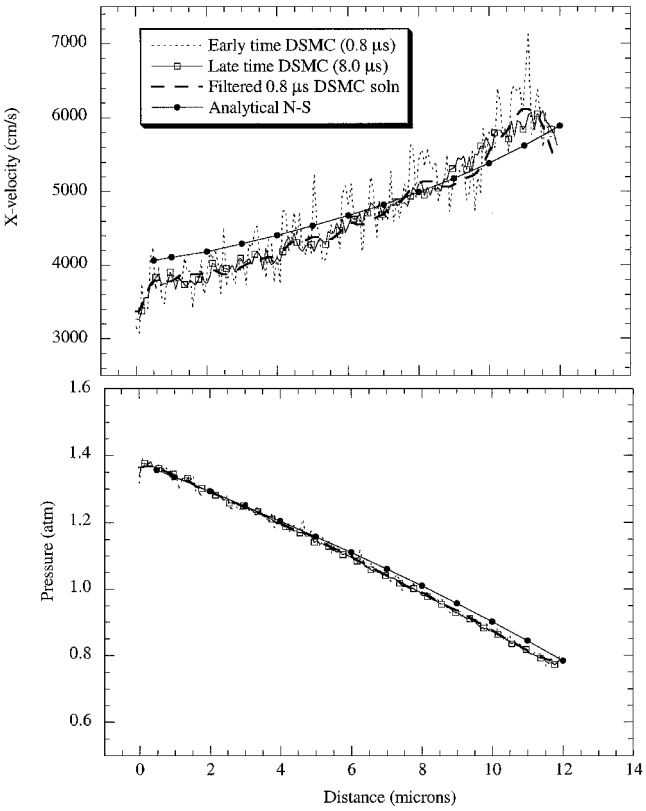

Fig. 9 Axial velocity and pressure along the longitudinal centerline for a low-Knudsen-number flow (N_2, V_{inflow} = 10,000 cm/s, and Kn = 0.05).

Figure 11 compares the results of the noisy DSMC (after 1000 μs of physical time), the filtered DSMC, and the analytic solution from Eqs. (8–10) for the $V_{\text{inflow}} = 200$ cm/s case. The pressure drop is linear, and the temperature in the microchannel is essentially constant. Figure 11 shows that the filtered V_x approximately follows the mean of the noisy V_x from the DSMC solution. Although the DSMC results for pressure and temperature are much less noisy than that for V_x, the filtered pressure and temperature are completely smooth and follow the same trends as those resulting from the DSMC calculation. Figure 11 also shows that the filtered DSMC solution closely follows the trends predicted by the analytical solution.

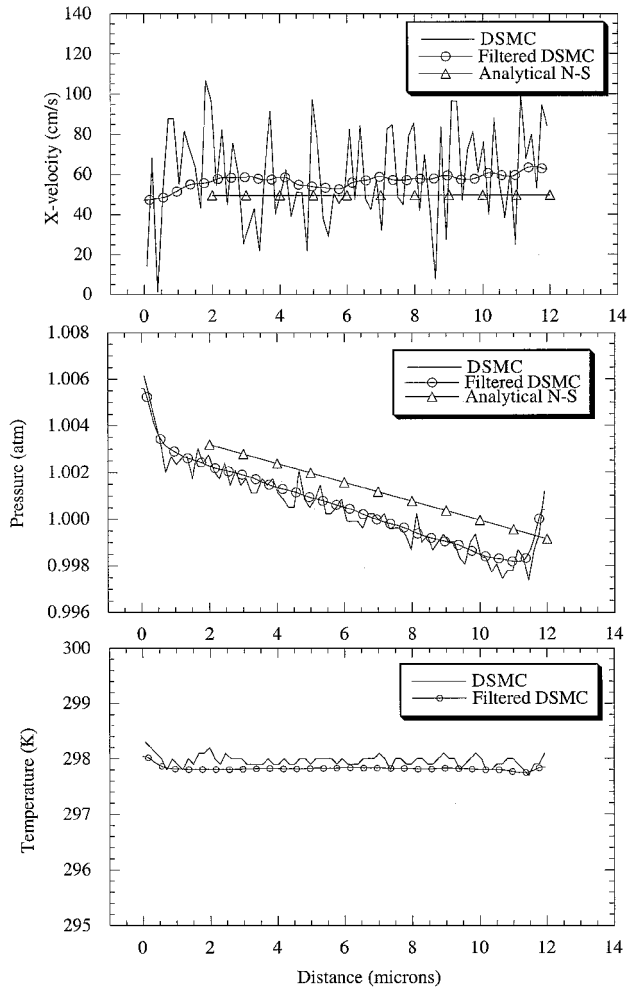

Fig. 11 Comparison between a late-time DSMC solution, filtered DSMC solution, and analytical solution to the Navier–Stokes equations for V_{inflow} = 200 cm/s case.

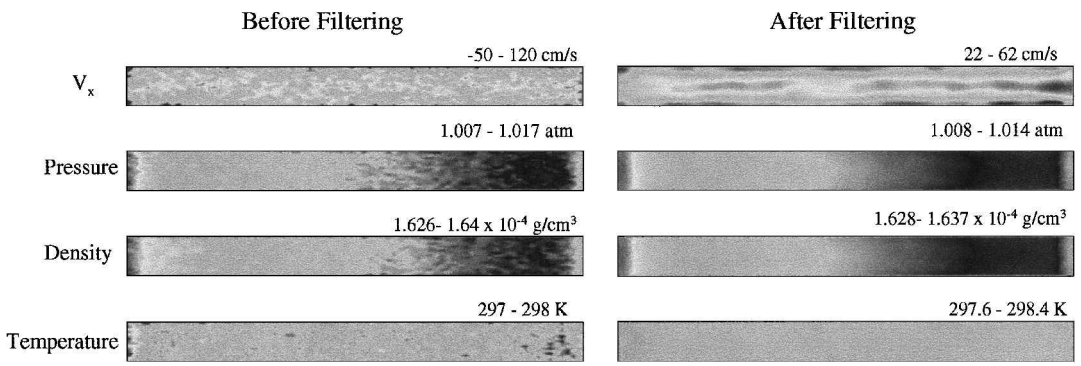

Fig. 10 V_{inflow} = 200 cm/s case, before and after FCT filtering.

Summary

We have discussed a promising approach for the removal of statistical noise from DSMC solutions and applied and tested this approach for microchannel flows with high and low velocities and high and low Knudsen numbers. The objective of this work was not to compute a simple microflow, but to demonstrate a generic method of nonlinear filtering that could be used for a wide variety of DSMC problems. One of the advantages of this approach is that it is a postprocessing operation and does not affect the DSMC calculation as it is running. This means that this nonlinear filtering method can be applied to any DSMC solution, including those with complex geometries, the presence of many interacting species with chemical reactions, varied and complicated boundary conditions, and time-dependent solutions.

During the process of developing this method, we tried a number of different ways to filter the DSMC results. These variations are outlined here:

1) As described earlier, one pass through the FCT filter was defined as operating on the DSMC data once in the x direction and once in the y direction. In addition, we tried filtering just in the x direction or just in the y direction. Neither approach did as well as filtering in both directions.

2) We also tried alternating the order of the direction of the filtering, that is, x–y and then y–x. This did nothing to improve the solution.

3) For all of the results presented, we convected the filtered quantity with zero velocity. We also tried adding in some small velocity, either a δv_x or a δv_y, then filtering, and then subtracting out the added velocity. This did not improve the solution any further.

4) We tried averaging multiple noisy solutions of the same problem before filtering, which did not help to remove statistical noise.

5) The flux-limiting procedure allows for some residual diffusion to be left in the solution. We tried using more or less residual diffusion in the flux-limited solution and found that the best results were obtained when a very small amount of residual diffusion is left in. Other approaches that will be examined in the future include changing the flux limiter in the algorithm and combining this with more or less residual diffusion.

A problem that frequently arises when simulating long microchannels is the requirement to use a large number of simulated particles. This is because there are many computational cells in a long microchannel, and if too few particles are used per cell, the statistical noise level increases. As we have shown, this could be removed to some extent by filtering with FCT.

In summary, we have shown that the nonlinear filtering technique, in this case using FCT, is able to extract a solution from a noisy DSMC calculation. It has significant potential for DSMC computations used in physical regimes where statistical noise must be eliminated. It is also possible that such filtering may be more generally applied to accelerate the solution of other types of Monte Carlo computations, and this is a topic for further investigation.

Acknowledgments

This project is supported by the Defense Advanced Research Projects Agency Design for Mixed Technology Integration Program. Computing resources were supplied by the Department of Defense High Performance Computing Modernization Program. The authors wish to thank Michael Macrossan for his insightful discussions on nonlinear filtering.

References

[1]Bird, G. A., *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Clarendon, Oxford, 1994, pp. 3, 40, 41.

[2]Oran, E. S., Oh, C. K., and Cybyk, B. Z., "Direct Simulation Monte Carlo: Recent Advances and Applications," *Annual Review of Fluid Mechanics*, Vol. 30, 1998, pp. 403–441.

[3]Oh, C. K., Oran, E. S., and Cybyk, B. Z., "Microchannel Flow Computed with the Direct Simulation Monte Carlo–Monotonic Lagrangian Grid," AIAA Paper 95-2090, Jan. 1995.

[4]Yasuhara, M., Nakamura, Y., and Takanaka, J., "Monte Carlo Simulation of Flow into Channel with Sharp Leading Edge," *Rarefied Gas Dynamics: Theoretical and Computational Techniques*, edited by E. P. Muntz, D. P. Weaver, and D. H. Campbell, Vol. 118, Progress in Astronautics and Aeronautics, AIAA, Reston, VA, 1989, pp. 582–596.

[5]Piekos, E. S., and Breuer, K. S., "Direct Simulation Monte Carlo Modeling of Micromechanical Devices," AIAA Paper 95-2089, Jan. 1995.

[6]Nguyen, T. X., Oh, C. K., Sinkovits, R. S., Anderson, J. D., and Oran, E. S., "Simulations of High Knudsen Number Flows in a Channel–Wedge Configuration," *AIAA Journal*, Vol. 35, No. 9, 1997, pp. 1486–1492.

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

[8]Mavriplis, C., Ahn, J. C., and Goulard, R., "Heat Transfer and Flowfields in Short Microchannels Using Direct Simulation Monte Carlo," *Journal of Thermophysics and Heat Transfer*, Vol. 11, No. 4, 1997, pp. 489–496.

[9]Arkilic, E. B., and Breuer, K. S., "Gaseous Flow in Small Channels," AIAA Paper 93-3270, July 1993.

[10]Beskok, A., and Karniadakis, G. E., "Simulation of Heat and Momentum Transfer in Complex Micro-Geometries," *Journal of Thermophysics and Heat Transfer*, Vol. 8, No. 4, 1994, pp. 647–655.

[11]Arkilic, E. B., Breuer, K. S., and Schmidt, M. A., "Gaseous Flow in Microchannels," *Application of Microfabrication to Fluid Mechanics*, FED-Vol 197, American Society of Mechanical Engineers, New York, 1994, pp. 57–66.

[12]Beskok, A., and Karniadakis, G. E., "A Model for Flows in Channels, Pipes, and Ducts at Micro and Nano Scales," *Microscale Thermophysical Engineering*, Vol. 3, No. 1, 1999, pp. 43–77.

[13]Harley, J. C., Huang, Y., Bau, H. H., and Zemel, J. N., "Gas-Flow in Microchannels," *Journal of Fluid Mechanics*, Vol. 284, 1995, pp. 257–274.

[14]Cai, C. P., Boyd, I. D., Fan, J., and Candler, G. V., "Direct Simulation Methods for Low-Speed Micro-Channel Flows," *Journal of Thermophysics and Heat Transfer*, Vol. 14, No. 3, 2000, pp. 368–378.

[15]Pan, L. S., Liu, G. R., Khoo, B. C., and Song, B., "A Modified Direct Simulation Monte Carlo Method for Low-Speed Microflows," *Journal of Micromachining and Microengineering*, Vol. 10, 2000, pp. 21–27.

[16]Bird, G. A., "Monte Carlo Simulation in an Engineering Context," *Rarefied Gas Dynamics*, edited by S. S. Fisher, Vol. 74, Progress in Astronautics and Aeronautics, AIAA, New York, 1981, pp. 239–255.

[17]Boris, J. P., and Book, D. L., "Flux-Corrected Transport I: SHASTA—A Fluid Transport Algorithm that Works," *Journal of Computational Physics*, Vol. 11, 1973, pp. 38–69.

[18]Boris, J. P., Landsberg, A. M., Oran, E. S., and Gardner, J. H., "LCPFCT-A Flux-Corrected Transport Algorithm for Solving Generalized Continuity Equations," U.S. Naval Research Lab., Memorandum Rept. 6410-93-7192, Washington, DC, April 1993, URL: http://www.lcp.nrl.navy.mil/lcpfct/.

[19]Oran, E. S., and Boris, J. P., *Numerical Simulation of Reactive Flow*, 2nd ed., Cambridge, Univ. Press, New York, 2001, pp. 235–240, 247–254.

[20]Hirsch, C., *Numerical Computation of Internal and External Flows, Volume 2: Computational Methods for Inviscid and Viscous Flows*, Wiley, New York, 1990, pp. 536–556.

[21]Boris, J. P., Grinstein, F. F., Oran, E. S., and Kolbe, R. L., "New Insights into Large Eddy Simulation," *Fluid Dynamics Research*, Vol. 10, No. 4–6, 1992, pp. 199–228.

[22]Fureby, C., and Grinstein, F. F., "Monotonically Integrated Large Eddy Simulation of Free Shear Flows," *AIAA Journal*, Vol. 37, No. 5, 1999, pp. 544–556.

[23]Arkilic, E. B., "Measurement of the Mass Flow and Tangential Momentum Accommodation Coefficient in Silicon Micromachined Channels," Ph.D. Dissertation, Fluid Dynamics Research Lab., TR97-1, Massachusetts Inst. of Technology, Cambridge, MA, Jan. 1997.

G. M. Faeth
Editor-in-Chief