Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 225 (2007) 1159–1174
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A modular particle–continuum numerical method for hypersonic non-equilibrium gas flows

T.E. Schwartzentruber *, L.C. Scalabrin, I.D. Boyd

University of Michigan, Department of Aerospace Engineering, 1320 Beal Avenue, Ann Arbor, MI 48109, USA

Received 31 July 2006; received in revised form 4 December 2006; accepted 24 January 2007
Available online 3 February 2007

Abstract

A modular particle–continuum (MPC) numerical method for steady-state flows is presented which solves the Navier–Stokes equations in regions of near-equilibrium and uses the direct simulation Monte Carlo (DSMC) method to simulate regions of non-equilibrium gas flow. Existing, state-of-the-art, DSMC and Navier–Stokes solvers are coupled together using a novel modular implementation which requires only a limited number of additional hybrid functions. Hybrid functions are used to adaptively position particle–continuum interfaces and update boundary conditions in each module at appropriate times. The MPC method is validated for 2D flow over a cylinder at various hypersonic Mach numbers where the global Knudsen number is 0.01. For the cases considered, the MPC method is verified to accurately reproduce DSMC flow field results as well as local particle velocity distributions up to 2.2 times faster than full DSMC simulations.

MSC: 65C05; 82C80

Keywords: Direct simulation Monte Carlo; DSMC; Hybrid particle–continuum; Hypersonics; Non-equilibrium flow; Rarefied flow; Re-entry vehicles

1. Introduction

Hypersonic vehicles travelling at high altitudes experience conditions ranging from highly rarefied to fully continuum flow. Even within a mostly continuum flow, there may be local regions of rarefied (or *non-equilibrium*) flow generated by rapid expansion in the wake of a vehicle as well as by strong gradients in shock waves and boundary layers. Separated flow caused by shock-boundary layer interactions can be greatly influenced by non-equilibrium effects as can the flow over sharp leading edges. Ultimately, it is the flow conditions near the vehicle surface and in the wake that determine the drag and heat transferred to the vehicle and its payload. Thus it is important that these regions be simulated accurately using an appropriate physical model. For

* Corresponding author. Tel.: +1 734 994 4924.
E-mail addresses: schwartt@umich.edu (T.E. Schwartzentruber), lscalabr@umich.edu (L.C. Scalabrin), iainboyd@umich.edu (I.D. Boyd).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.01.022

example, the continuum Navier–Stokes (NS) equations assume collisional near-equilibrium. More specifically, the velocity distribution function (vdf) of gas particles at every point in the flow is assumed to deviate only slightly from an equilibrium (Maxwell–Boltzmann) distribution. This perturbed equilibrium distribution is of a special form called a Chapman–Enskog distribution. The NS equations deal only with the macroscopic representations of this distribution, such as the density, temperature and bulk velocity of the gas. The heat and momentum transferred to the vehicle is then calculated using gradients of these macroscopic values at the vehicle surface. However, in high-speed, high-altitude environments there are regions in the flow which do not reach collisional equilibrium. For example, the rapid flow expansion around a blunt body creates a low density wake region where particles collide less frequently. Likewise, while traversing strong shock waves, thin boundary layers, and sharp leading edges, high-speed gas particles may not undergo a sufficient number of collisions to fully "equilibrate". As a result, gas particles in these non-equilibrium regions do not have a Chapman–Enskog velocity distribution and thus can not be accurately characterized by the NS equations. Macroscopically, such non-equilibrium flow conditions may result in both temperature jump and velocity slip at the vehicle surface as well as changes in flow separation regions; all of which can significantly influence the heat and momentum transfer to a vehicle.

1.1. Deterministic solution of the Boltzmann transport equation

The fundamental governing equation for dilute gases is the Boltzmann transport equation (BTE), presented in Ref. [1] in the following form:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = \Delta[Q(f)]. \tag{1}$$

This equation describes the evolution of the single-particle distribution function, $f(\mathbf{x}, \mathbf{v}, t)$ which represents the molecular density of particles with position $\mathbf{x}(x, y, z)$ and particle-velocity $\mathbf{v}(v_x, v_y, v_z)$ at time t. In Eq. (1), f is also referred to as the velocity distribution function (vdf). The operator, $\Delta[Q(f)]$, describes the rate of change of f due to binary collisions among particles. Unlike calculation of the other terms in Eq. (1), $\Delta[Q(f)]$ involves integrating over all velocity space in order to account for collisions between particles of all velocities. In contrast with the NS equations, since no assumption about the form of f is made, the BTE is physically accurate for dilute gases under all conditions ranging from continuum to free-molecular flow. For this reason, various researchers have used the BTE as the basis for numerical simulations of gas flows which contain a mixture of continuum and rarefied conditions. For example Kolobov et al. propose a "unified" continuum/kinetic method [2] where the collision-less step of the method (the left-hand-side of Eq. (1)) is identical in both continuum and non-equilibrium regions. Only the calculation of the collision integral, $\Delta[Q(f)]$, is different. In non-equilibrium regions a full integration of $\Delta[Q(f)]$ over all velocity space is performed using various numerical schemes such as a Monte Carlo evaluation. In the continuum regions, calculation of the collision integral is replaced by a "Maxwellization" of the distribution function, f. This simplification is reported as being more than an order of magnitude less computationally expensive than a Monte Carlo evaluation of the integral using 100 random trials [2]. As a result, the computational time spent on the continuum regions is insignificant compared to that spent on the non-equilibrium regions. This algorithm was recently used to simulate flow about a sphere as well as for more complex flow about re-entry capsule and shuttle geometries [3]. Another approach proposed by Le Tallec and Mallinger is to use a BTE solver in non-equilibrium regions and a NS solver for cells in continuum regions. Information transfer is made consistent by matching the half-fluxes at the interface between the BTE and NS solvers. The authors apply this method to 2D hypersonic flow of a monatomic gas over a flat plate and ellipse [4]. Although these methods are more efficient than solving the BTE throughout the entire flow field, it is important to note that solving the BTE is computationally intensive to begin with. Even for a monatomic gas with no internal degrees of freedom a six-dimensional computational grid is required, consisting of three spatial and three velocity dimensions. If the gas is not monatomic, even more dimensions are needed for internal degrees of freedom such as rotational and vibrational energies. For 3D, multi-species flows, this high dimensionality becomes extremely computationally demanding and spatial resolution is severely limited. Faced with these computational constraints, one must also decide how to set the bounds and resolution of velocity space as well as rotational and vibrational energy space.

1.2. Coupling particle and continuum methods

A more popular method for simulating rarefied, non-equilibrium flow is the direct simulation Monte Carlo (DSMC) method developed by Bird [1]. The DSMC method effectively solves the BTE (Eq. (1)) by directly simulating the movement and collision of particles within the gas and with the vehicle walls. This allows for temperature jump and velocity slip to be natural results of a DSMC simulation and the heat and momentum flux to the vehicle is actually calculated by particle collisions with the surface. In a DSMC simulation, the trajectories of a large number of simulated particles are followed simultaneously through a grid of computational cells. For each iteration, particles are first moved along their trajectories without colliding, after which particles residing within the same cell are randomly selected for a collision process. A major limitation of DSMC is that in order for this collision process to be physically accurate, the cell size must be on the order of the mean free path (λ) and each DSMC cell must contain at least 20 simulated particles [1]. As a result, 2D and 3D DSMC simulations can require prohibitively high numbers of computational cells and therefore simulated particles, especially in regions where λ is very small. However, it is precisely in these regions where collisional equilibrium is reached and the continuum equations can be solved. For example, Pareschi and Caflisch present a novel method [5] where f is represented as a combination of a Maxwellian distribution (M) and a non-equilibrium distribution of DSMC particles (g); $f = \beta M + (1-\beta)g$. In regions approaching collisional equilibrium particles are removed from the DSMC simulation and more weight is transferred to the Maxwellian distribution. Thus, in the continuum limit, the method essentially becomes an Euler solver.

An alternate approach taken by many researchers is to directly couple DSMC with traditional NS solvers. Unlike numerical methods based on the BTE, this approach involves coupling two entirely different methods, one deterministic in nature and one statistical. Ref. [6] presents a discussion of the major considerations involved as well as a summary of published work on such coupled schemes. One important aspect is the transfer of information between particle and continuum cells and the inherent statistical scatter in that information. Information transfer is typically handled using one of the two methods depicted in Fig. 1. Flux-based coupling, Fig. 1(a), involves calculating the fluxes of mass, momentum and energy at the interface according to the particle cell (F_P), and according to the continuum cell (F_C). F_P is calculated by tracking individual simulation particles as they cross the interface whereas F_C is extrapolated using macroscopic gradients in the continuum domain. F_P is then used to update the continuum solution while at the same time F_C is used to create a distribution of particles on the interface which propagate into the particle simulation. On the other hand, state-based coupling (Fig. 1(b)) temporarily averages particle information to obtain a macroscopic state on one side of the interface and at the same time generates a distribution of particles from a macroscopic state on the other side of the interface. In this way, the flux into both the continuum domain and particle domain is handled exclusively by its respective solver through standard boundary procedures. Flux-based coupling was used by Hash and Hassan to couple the DSMC method with a NS solver for simulations of Couette flow [7] and hypersonic flow over a blunted cone [8]. More recently, Wijesinghe et al. also used flux-based coupling

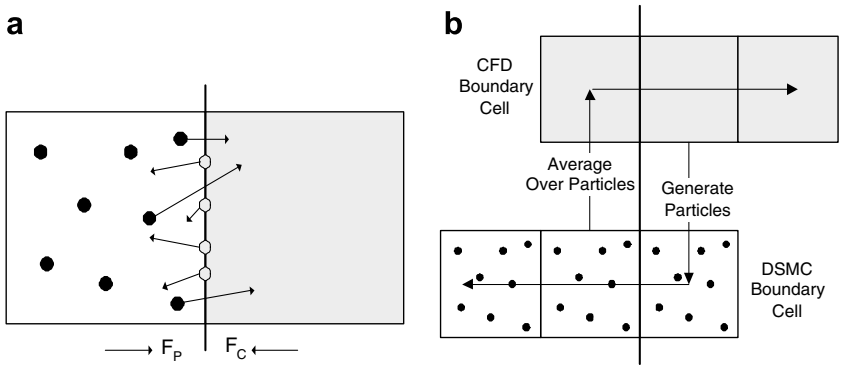

Fig. 1. Typical hybrid coupling procedures. (a) Flux-based coupling. (b) State-based coupling.

to embed a DSMC solver in the finest level of an adaptive mesh and algorithmic refinement (AMAR) scheme for the Euler equations [9]. However, when averaging particle information, the statistical scatter involved in determining F_P in Fig. 1(a) is much higher than that associated with averaging to obtain a macroscopic state [10] as seen in Fig. 1(b). Additionally, as discussed in Ref. [11], fluxes normal to the interface may become very small in regions where the interface is aligned with the flow direction. As a result, the statistical scatter inherent in the flux quantities becomes even more pronounced; a significant disadvantage of flux-based coupling. Roveda et al. used state-based coupling to combine the DSMC method with an Euler solver to simulate moving 1D shock waves [12] and 2D unsteady slit flow [13]. Although use of state-based coupling involved less statistical scatter, their method was time-accurate and particle information was averaged at each time step. This significantly constrained the number of samples used for the average. In order to deal with this, the authors employed a novel algorithm to effectively "clone" particles near the interface in order to reduce the statistical scatter transferred to the continuum domain.

1.3. A modular particle–continuum approach for steady-state flows

The numerical method outlined in this article focuses on steady-state, hypersonic flows. A similar method has been successfully tested for 1D normal shock waves [14] and is here extended and modified for 2D flows. The high velocities associated with hypersonic flow greatly reduce the influence of statistical scatter making DSMC an efficient numerical method in this regime. By further restricting the method to steady-state flows, this article will show that *temporal* averaging used in conjunction with state-based coupling is able to effectively manage statistical scatter. Focusing on steady-state flows also allows use of an implicit NS solver, enabling time-scale decoupling. It should be noted that low-speed and/or unsteady flows have none of these advantages. Thus formulating a general high-speed/low-speed, steady/unsteady hybrid method is a significant challenge which requires further research. Incorporating the above ideas, this article presents a novel modular implementation. The algorithm combines *existing* state-of-the-art DSMC and NS solvers (virtually un-modified) into one modular particle–continuum (MPC) numerical method. This article will first provide a brief overview of the DSMC and NS solvers used. Next, the positioning of particle–continuum interfaces, the control of statistical scatter, and the method of information transfer will be described. State-based coupling allows for information transfer to be handled by the existing boundary procedures used by both DSMC and NS solvers. These components will then be incorporated into a modular numerical cycle which ensures the overall solution progresses towards the correct physical result. The MPC method will then be validated for hypersonic flow over a 2D cylinder by comparison with NS and pure DSMC results and the computational efficiency of the method will be discussed.

2. Numerical models and flow conditions

The MPC code described in this article divides the flow field into particle regions where the DSMC method is used and continuum regions where the NS equations are solved. Existing, state-of-the-art codes are used which have been extensively validated. Complete details for each code can be found in the references.

Particle regions are simulated using MONACO [15], a general, cell-based implementation of the DSMC method [1] which statistically simulates the Boltzmann transport equation (Eq. (1)). The variable hard sphere (VHS) collision model is employed which results in the following macroscopic viscosity model [1]:

$$\mu = \mu_{\text{ref}} \left(\frac{T}{T_{\text{ref}}} \right)^{\omega}, \quad \mu_{\text{ref}} = \frac{15\sqrt{\pi m k T_{\text{ref}}}}{2\pi d_{\text{ref}}^2 (5 - 2\omega)(7 - 2\omega)}. \tag{2}$$

All numerical results presented in this article are for diatomic nitrogen with a reference diameter of $d_{\text{ref}} = 4.17 \times 10^{-10}$ m at $T_{\text{ref}} = 273$ K. The power law exponent, ω, is set equal to 0.75, m is the mass of an N_2 molecule, and k is the Boltzmann constant. MONACO employs the variable rotational energy exchange probability model of Boyd [16] where the reference temperature for rotational energy exchange is specified as 91.5 K and the maximum rotational collision number as 18.1. Energy transfer to vibrational modes is not considered.

Continuum regions are simulated using the LeMANS code [17]. For the results presented in this article, it is assumed that rotational and translational energy modes can be described by a single temperature T in continuum regions. The vibrational energy mode is not considered. The resulting governing equations are the well-known, 2D laminar, compressible, Navier–Stokes (NS) equations. It should be noted that the NS equations result directly from the BTE (Eq. (1)) when f is assumed to be a Chapman-Enskog distribution function [1]. The viscosity in the NS solver is modeled using Eq. (2) in order to match exactly the viscosity model used in DSMC. This is important not only at particle–continuum interfaces (for information transfer) but also whenever comparing DSMC and NS flow field results and surface properties. LeMANS solves this set of equations using a finite-volume formulation. The inviscid fluxes between the mesh volumes are discretized using a modified form of the Steger–Warming Flux Vector Splitting [18] which is less dissipative than the original form. The modified form is thus adequate to calculate boundary layers and the scheme switches back to the original form of Steger–Warming near shock waves. The viscous terms are calculated using the values of properties at the cell centers and at the nodes. The time integration is performed using a point-implicit method. Specific details of the numerical method are contained in Ref. [17].

High-speed flow about a 2D cylinder at Mach 3, 6 and 12, is simulated in cases denoted as **M3**, **M6** and **M12**, respectively. The free-stream gas is N_2 with a number density $n = 1.61 \times 10^{21}\ 1/\mathrm{m}^3$ and $T = 217.45$ K. This corresponds to an altitude of 70 km, where $p = 4.83$ Pa and $\rho = 7.48 \times 10^{-5}\ \mathrm{kg/m}^3$. The diameter of the cylinder is 8.0 cm resulting in a global Knudsen number of 0.01. Only the free-stream velocities differ for each case and are set to achieve the desired Mach numbers. The cylinder wall temperature is set to 300 K, 500 K and 1000 K for cases **M3**, **M6** and **M12**, respectively. The DSMC solver assumes diffuse reflection and full thermal accommodation at the cylinder wall whereas the NS solver employs no-slip conditions. The same mesh is used for all NS, DSMC and MPC simulations of all cases. The mesh is structured and consists of 600 evenly spaced cells along the surface of the cylinder and 300 cells in the radial direction. In the fore-body, the cell size is clustered towards the cylinder surface. For all cases the cell size is verified to be less than the local mean-free-path throughout the domain. This constraint is a requirement of the DSMC method and is more than sufficient for an accurate NS simulation. The reference particle weight is set to obtain at least 20 particles per cell in the wake region which results in approximately 50 particles per cell, on average, in the fore-body flow. No particle weighting, nearest-neighbor collision routine, or sub-cell sampling is used. Constant DSMC time-steps of 5×10^{-8} s are used for both the **M3** and **M6** cases, while a time-step of 2×10^{-8} s is used for case **M12**. These time-steps are less than one-half of the shortest mean-free-time anywhere in each simulation.

3. Interface location and information transfer

The efficiency of a hybrid particle–continuum method comes from restricting use of the DSMC method to only that portion of the flow field where large non-equilibrium effects are felt. Ultimately, the particle–continuum *interface* must lie in a region which is accurately modelled by the continuum NS equations. Ensuring this is a vital aspect of any such hybrid method. The MPC method determines particle and continuum regions using a local continuum breakdown parameter called the gradient-length Knudsen number

$$Kn_{\mathrm{GL}-Q} = \frac{\lambda}{Q}|\nabla Q|, \tag{3}$$

where Q represents a local flow quantity such as density (ρ), temperature (T), or bulk velocity magnitude ($|\mathbf{V}| = \sqrt{u^2 + v^2}$), and λ is the local mean-free-path. The final value used to quantify the degree of local continuum breakdown is then taken as the maximum for all flow quantities,

$$Kn_{\mathrm{GL}} = \max(Kn_{\mathrm{GL}-\rho}, Kn_{\mathrm{GL}-T}, Kn_{\mathrm{GL}-|\mathbf{V}|}). \tag{4}$$

Previous research [14,19,20] has validated the use of this parameter and recommended values of $Kn_{\mathrm{GL}} > 0.05$ as signifying continuum breakdown for representative hypersonic flows. The MPC method begins with a NS solution to the problem and uses it to initially predict where the non-equilibrium regions are located. Fig. 2 (top) plots the value of Kn_{GL} from Eq. (4) applied to an initial NS solution of the **M6** case. The continuum breakdown parameter is seen to be largest in the bow shock, fore-body boundary layer, and in the near wake

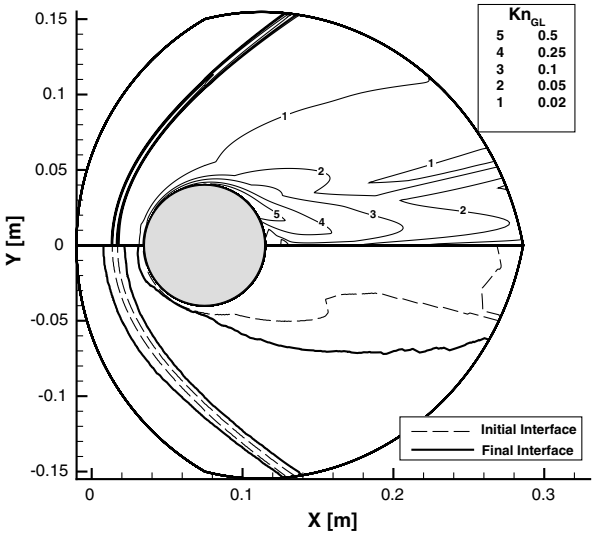

Fig. 2. Continuum breakdown and interface locations.

region. Cells in which $Kn_{GL} > 0.05$ are labelled as DSMC cells and the remainder are labelled as NS cells. This results in the initial interface (dashed line) shown in the bottom of Fig. 2. This interface corresponds to the $Kn_{GL} = 0.05$ contour in the top of Fig. 2 after a simple smoothing algorithm has been applied to the interface. As the MPC solver iterates, the solution may change significantly from the initial continuum solution towards the correct non-equilibrium solution. Thus, the interface locations must adapt during the MPC simulation. As an example, the final interface location for the **M6** case is shown in the bottom of Fig. 2 by the solid line.

The MPC method uses state-based coupling to transfer information between particle and continuum regions as detailed in Fig. 3. After application of the breakdown parameter (Kn_{GL}), the particle region is immediately extended by a few extra cells into the continuum domain to create an overlap region. The need for this overlap region will be discussed shortly. Next, one row of NS boundary cells and two rows of DSMC boundary cells are initialized as seen in Fig. 3. Now that all regions of NS and DSMC cells, including boundary cells, have been defined, the regions must be coupled by transferring information across the interface. When using state-based coupling this simply involves updating the boundary conditions of each solver. In this way, information transfer into both the particle and continuum regions is handled through existing boundary procedures already used by both solvers. The details involved in setting both DSMC and NS boundary conditions, including the reduction of statistical scatter can be found in Ref. [14]. The DSMC boundary cells are

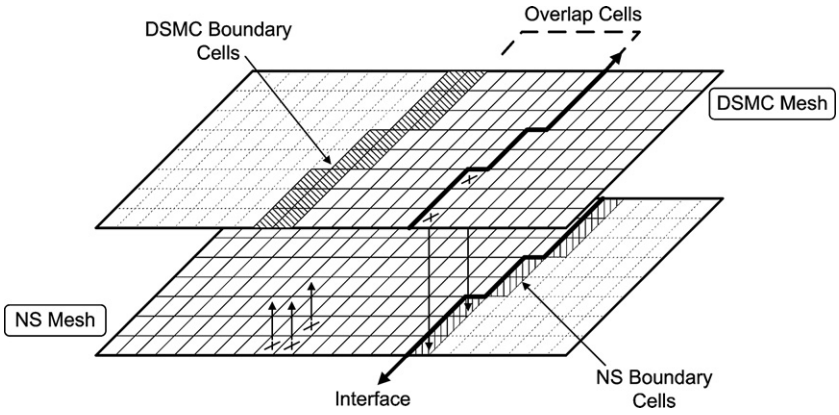

Fig. 3. MPC coupling procedure.

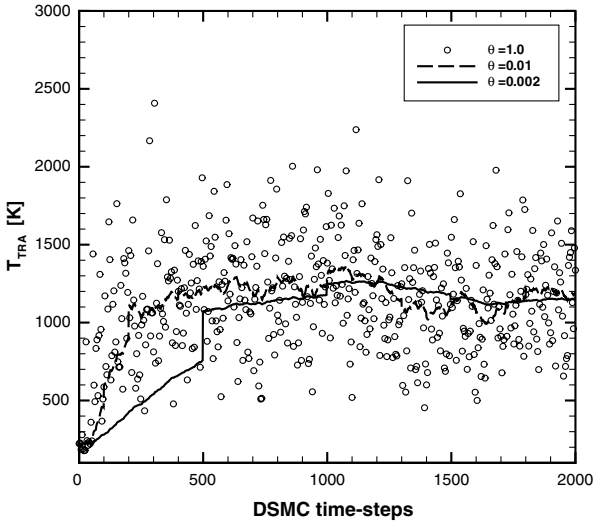

Fig. 4. Sub-relaxation average.

continually filled with particles consistent with the flow properties in the corresponding NS cell using a Chapman-Enskog velocity distribution [21]. As particles in the DSMC region interact and their distributions evolve in time, the MPC method also tracks the *macroscopic* variation in each DSMC cell. In order to provide these averaged properties with minimal statistical scatter, a mixture of spatial and temporal averaging is used. Specifically, the MPC method uses the sub-relaxation technique proposed by Sun and Boyd [22], where the formula for the *temporal* average at time-step j of the desired macroscopic properties ($\overline{Q_j}$) is calculated in each cell as

$$\overline{Q_j} = (1-\theta)\overline{Q_{j-1}} + \theta Q_j. \tag{5}$$

Here, θ is the small weight applied to each new cell-averaged property and $(1-\theta)$ is a larger weight applied to the previous history. For example, Fig. 4 shows the translational temperature variation in a DSMC cell located just upstream of the bow shock in the **M6** case. The cell is originally at the pre-shock state ($T = 217.45$ K). However, as the simulation proceeds, the shock thickens, and the cell ends up inside the shock ($T = 1100$ K). The symbols in Fig. 4 represent the temperatures calculated using only the spatial average in a DSMC cell at each time-step. The resultant statistical scatter is seen to be larger than the macroscopic variation itself. Fig. 4 shows that introducing a temporal average via the sub-relaxation technique with $\theta = 0.002$ eliminates the scatter while a correction [14,22] improves the time-lag. For all results presented in this article, a value of $\theta = 0.001$ is used to provide low-scatter, macroscopic quantities in each DSMC cell at each time-step. When desired, these averaged DSMC properties can then be used to update boundary conditions for the NS solver.

4. Modular numerical cycle

The final aspect of a hybrid method is the numerical cycle which determines *when* to transfer information between particle and continuum regions. The most important feature of the MPC numerical cycle is that before any information is transferred to the continuum region, interfaces must first be positioned in near-equilibrium regions.

4.1. Interface adaptation

As previously described, the MPC cycle begins with a continuum NS solution to the problem. The continuum breakdown parameter, Kn_{GL}, is then applied to initially setup particle and continuum regions and

an overlap region is extended. Next, particles are generated throughout the newly created particle region. These particles are sampled from a Chapman–Enskog distribution [21] consistent with the NS solution. The MPC cycle then proceeds to iterate the DSMC solver in particle regions without any coupling to the continuum regions. The results for translational temperature along the stagnation line for the **M6** case are shown in Fig. 5(a). Here the DSMC regions have iterated for 2000 time-steps and the particle distributions have evolved significantly away from their initial continuum distributions towards the correct non-equilibrium distributions. This important behavior was previously observed in 1D simulations of normal shock waves [14]. The solid MPC temperature profile in Fig. 5(a) is the result of the sub-relaxation average used in each DSMC cell. In fact, the temperature variation in the cell at the pre-shock interface, where the temperature is seen to vary from 217 K to 1100 K, was plotted previously in Fig. 4. In Fig. 5(a), since the interfaces remain at their original positions, the post–shock interface now lies inside the temperature overshoot (a non-equilibrium region). If this state is used as a new NS boundary condition, Fig. 5(b) shows the change in the continuum solution at two times, t_1 and t_2. Clearly, the NS equations are not capable of predicting the tail of the temperature overshoot and instead predict a different post–shock state. This large error will eventually be transmitted back into the particle region via the DSMC boundary cells and the error may never leave the simulation.

Instead, the overlap region can be utilized to better position the interfaces before any information is transferred. In Fig. 5(a), significant flow gradients are now present in the overlap regions. Application of Kn_{GL} to the MPC profile will thus convert cells in the overlap region to pure-particle cells. In this case, the particle region would enlarge, a new overlap region would be extended further, and the process would repeat. At some point, the entire overlap region will be located in a near-continuum region (small flow gradients), and further application of Kn_{GL} will not alter the interface location. It should be noted that the interfaces can be updated far more frequently, effectively adapting *while* the DSMC region is proceeding to steady-state. Only when the interface stops moving, and is ensured to be in a near-continuum region, does the MPC method transfer information to the continuum domain.

Along the stagnation line in Fig. 5(a), the post–shock state for both continuum and particle solutions is identical and thus information transfer will not alter the NS boundary condition. However, along a 60° cut in the fore-body flow the gas immediately expands after passing through the shock. This expansion induces thermal non-equilibrium and as seen in Fig. 6 (for case **M12**) the post–shock state predicted by DSMC is now slightly different. Since significant energy remains ''frozen'' in the rotational mode, the translational temperature predicted by DSMC is lower than the single temperature predicted by the NS equations. It is impor-

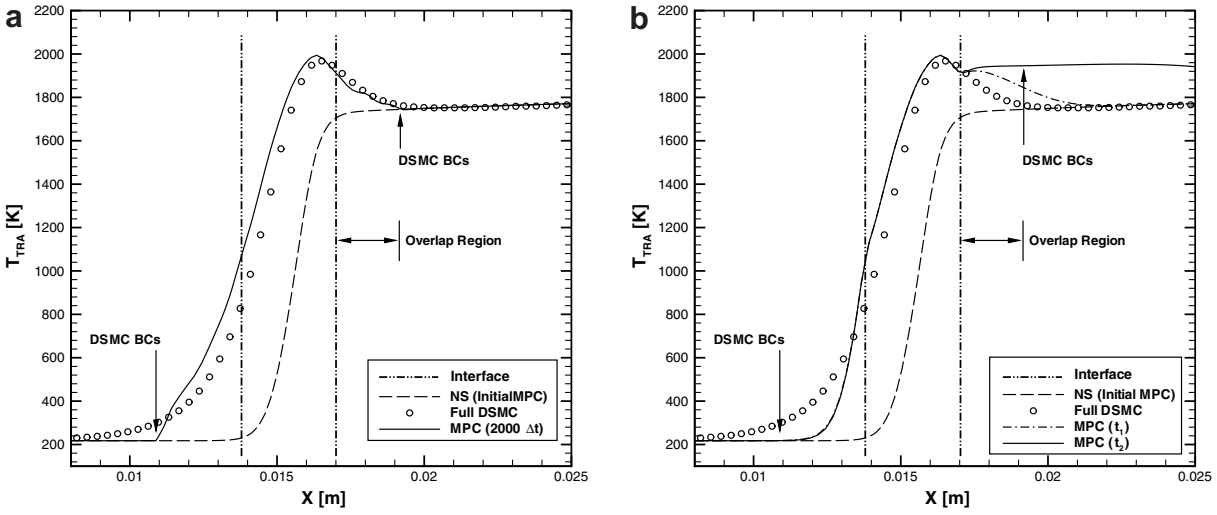

Fig. 5. Progression of the DSMC solution and interface adaptation (case **M6**). (a) T_{TRA} profile along stagnation line. (b) Effect of incorrect interface location.

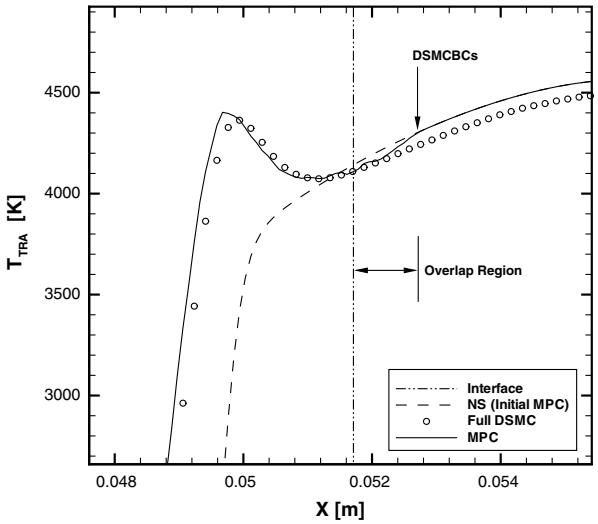

Fig. 6. T_{TRA} profile along a 60° cut (case **M12**).

tant to note how the incorrect boundary conditions imposed on the DSMC region have only a small influence on neighboring interior cells. This behavior was also observed for hybrid simulations of 1D shock waves [14]. Thus, as well as enabling the interfaces to adapt, the overlap region also serves to isolate the interior DSMC region from incorrect boundary conditions supplied from the NS solution. In Fig. 6, when information is transferred at the interface, the NS temperature boundary condition is now lower. This will eventually shift the entire profile in the NS region into better agreement with the full DSMC solution, and as a result, will then improve the DSMC boundary conditions.

It is important to note that the interface location is ultimately dependent on the definition and empirically-determined cutoff value for continuum breakdown. Although using Eq. (4) with a cutoff value of 0.05 works quite well for pre-shock, stagnation, and wake regions, it fails to correctly position the post–shock interface. When the cutoff value is lowered in an attempt to correct this, the parameter becomes far too sensitive in the other regions. The difficulty is not the cutoff value, but arises from thermal non-equilibrium in the post–shock region which the Kn_{GL} does not account for. As suggested in Ref. [14], a condition of thermal equilibrium is added to the breakdown parameter which now becomes

$$Kn_{\mathrm{GL}} = \max\left(Kn_{\mathrm{GL}-\rho}, Kn_{\mathrm{GL}-T}, Kn_{\mathrm{GL}-|\mathbf{v}|}, 5 \times \frac{T_{\mathrm{TRA}} - T_{\mathrm{ROT}}}{T_{\mathrm{ROT}}}\right). \tag{6}$$

In continuum (NS) regions, $T_{\mathrm{TRA}} = T_{\mathrm{ROT}} = T$, however in particle regions these temperatures may not be equal. Only the very strong thermal non-equilibrium in the shock wave needs attention. This is a compression region in which $T_{\mathrm{TRA}} > T_{\mathrm{ROT}}$, whereas the remainder of the flow field is in expansion non-equilibrium with $T_{\mathrm{TRA}} < T_{\mathrm{ROT}}$. In order to focus only on the shock region, the temperature difference in Eq. (6) is calculated without absolute value signs. Thus, in regions where $T_{\mathrm{TRA}} > T_{\mathrm{ROT}}$ by more than 1%, the breakdown parameter is given a value, $Kn_{\mathrm{GL}} > 0.05$. The continuum breakdown parameter expressed by Eq. (6), with a cutoff value of 0.05, is used for all MPC simulations presented in this article.

4.2. Cycle details

The above ideas and components are incorporated in the MPC numerical cycle which is completely detailed by the following steps:

(1) Obtain a continuum solution to the problem by solving the NS equations. Use the continuum breakdown parameter to initialize DSMC regions and generate particles consistent with the NS solution.

(2) Cycle the DSMC solver using the current NS solution for boundary conditions. Use the continuum breakdown parameter and an overlap region to adapt the interfaces without transferring any information to the NS regions.
(3) After the DSMC solution and interfaces stop changing, use the current DSMC solution to update the NS boundary conditions. Proceed to significantly converge the NS region.
(4) **IF** execution of step (2), using this new NS solution, does not modify the interface locations → continue to (5).
ELSE → return to (2).
(5) Since the DSMC solution and interface locations are no longer changing *and* the NS region is converged, steady-state has been reached. Lock the interfaces and cycle both the DSMC and NS solvers (coupling occasionally) until the DSMC scatter and NS residual fall below threshold values.

The implementation of step (1) has already been outlined in previous sections. The precise implementation of step (2) is detailed by the following steps:

(2a) Cycle the DSMC solver for one time-step with boundary conditions set using the current NS solution. Use the sub-relaxation average to determine macroscopic properties in each DSMC cell.
IF any flow quantity in any cell touching the interface has changed by more than 20% *or* the DSMC solver has cycled for $\frac{1}{\theta}$ time-steps → continue to (2b)
ELSE → return to (2a)
(2b) Apply the breakdown parameter to the current MPC profile and re-label cells as DSMC or NS.
(2c) Extend an overlap region and re-label DSMC and NS boundary cells.
(2d) Generate particles in any *newly-created* DSMC cells based on the corresponding state from the NS solution.
IF the number of DSMC cells has changed by more than 1% → return to (2a).
ELSE if the number of DSMC cells has changed by less than 1% → proceed to step (3).

For the results presented in this article, 10 overlap cells are used and θ is set to 0.001. Applying the breakdown parameter after every 20% change allows the interfaces to adapt *while* the DSMC solution is proceeding towards steady-state. If no significant changes are detected, the DSMC solver is allowed to iterate for $\frac{1}{\theta}$ time-steps at which point the next correction to the sub-relaxation average is made (see Fig. 4). This allows sufficient time for the DSMC solution to change further (in case it has not yet reached steady-state) before re-application of the breakdown parameter. Currently, in step (3), the state in each NS boundary cell is completely specified from the DSMC solution in both supersonic and subsonic regions of the flow. Although this is not found to cause any stability problems, it does prevent the NS solution from converging to machine zero. Instead, during step (3), 500 implicit time-steps are taken using a CFL value of 5.0. This converges the solution by roughly 3 orders of magnitude after which no visible changes in the NS regions are observed. Finally in step (5), after the interfaces are fixed and the DSMC solution has reached steady-state, the sub-relaxation average is no longer required. Instead, standard DSMC sampling is used to calculate the macroscopic flow quantities. Every 5000 DSMC time-steps, the sampled information is transferred to the NS regions which are converged for 100 implicit time-steps. This cycle repeats and continued sampling is able to reduce the statistical scatter. Thus, as the scatter in the DSMC solution is eliminated, so too is the scatter transferred to the NS regions. The result is a scatter-free DSMC solution in non-equilibrium regions which transitions to a converged NS solution in continuum regions. Upcoming sections will describe validation of the MPC method for 2D hypersonic flow about a cylinder.

5. Numerical validation

The MPC method will be shown capable of reproducing full DSMC simulation results for macroscopic flow field quantities as well as microscopic velocity distribution functions. As a reference, the final interface locations for each simulation are shown in Fig. 7. As the flow Mach number is increased, the size of the non-equilibrium regions simulated using DSMC increases. For case **M3**, the particle region terminates before the

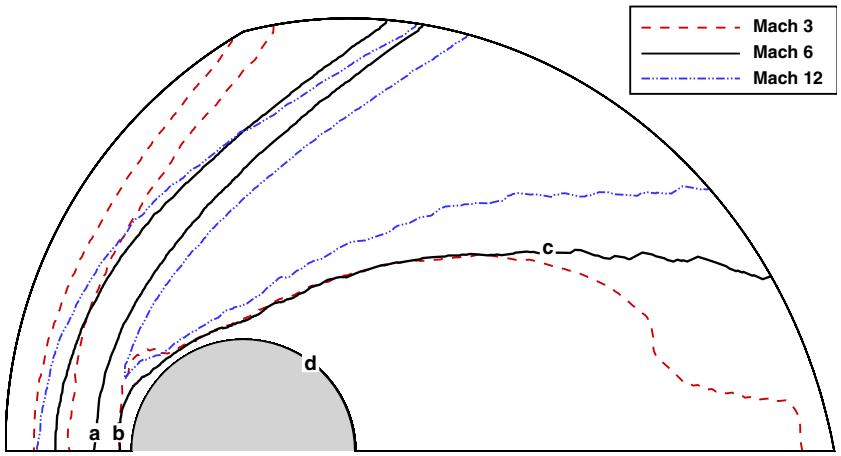

Fig. 7. Final MPC interface locations.

domain exit, whereas case **M12** requires a larger particle region where the shock and boundary layer particle regions actually merge in the fore-body flow.

5.1. Velocity distribution functions (vdfs)

The motivation behind a hybrid particle–continuum method is the fact that large regions in a DSMC simulation may contain particles with equilibrium distributions. Significant computational work may then be devoted to collisions between these particles which simply act to maintain their equilibrium distributions! In such near-equilibrium regions, the NS equations are fully capable of modelling the flow and contain all of the necessary information to describe the correct vdf. Since hybrid interfaces lie in near-equilibrium regions, the NS equations should be capable of generating the exact particle vdfs at the interface that are seen in a full DSMC simulation. In order to demonstrate this for the **M6** case, Fig. 8 plots the local vdfs at positions **a** through **d** (seen in Fig. 7) for both MPC and full DSMC simulations. Fig. 8(a)–(c) plot the distributions of particles generated in the DSMC boundary cells by the MPC method at locations **a**, **b** and **c**, respectively. These distributions are seen to be identical to the particle distributions measured in the same cell of a full DSMC simulation. As expected, in these regions of the flow, particles have near-equilibrium vdfs. Thus, if the interfaces seen in Fig. 7 are surrounded by the same particles present in full DSMC simulations, the interior particle solution should agree exactly with a full DSMC simulation. In order to demonstrate this, the vdf in a non-equilibrium region (location **d**) is plotted in Fig. 8(d). Indeed, the MPC method successfully reproduces the non-equilibrium vdf predicted by a full DSMC simulation. In order to highlight the necessity of a particle simulation in this region, the Chapman–Enskog vdf predicted by a full NS solution is also plotted in Fig. 8(d) for comparison. Clearly, velocity slip is present at this point on the surface and the shape of the vdf is quite far from equilibrium.

5.2. Flow field quantities

Despite using the DSMC method only in portions of the flow field, the MPC method successfully reproduces full DSMC results with little error. The largest error is found in the T_{TRA} field away from the surface in the post–shock continuum region. MPC and full DSMC solutions for the T_{TRA} field of the **M12** case are overlayed in Fig. 9. The solutions are in excellent agreement and further analysis finds that the largest error in any flow property is found in T_{TRA} and is limited to 3%, 2.5% and 1.5% for cases **M12**, **M6** and **M3**, respectively. The flow properties along a streamline (which passes through this region of maximum error) are plotted in Fig. 10(a) and (b). The thermal non-equilibrium responsible for this error can be seen in Fig. 10(a). Inside the particle region encompassing the shock, the MPC method precisely reproduces full DSMC results for both

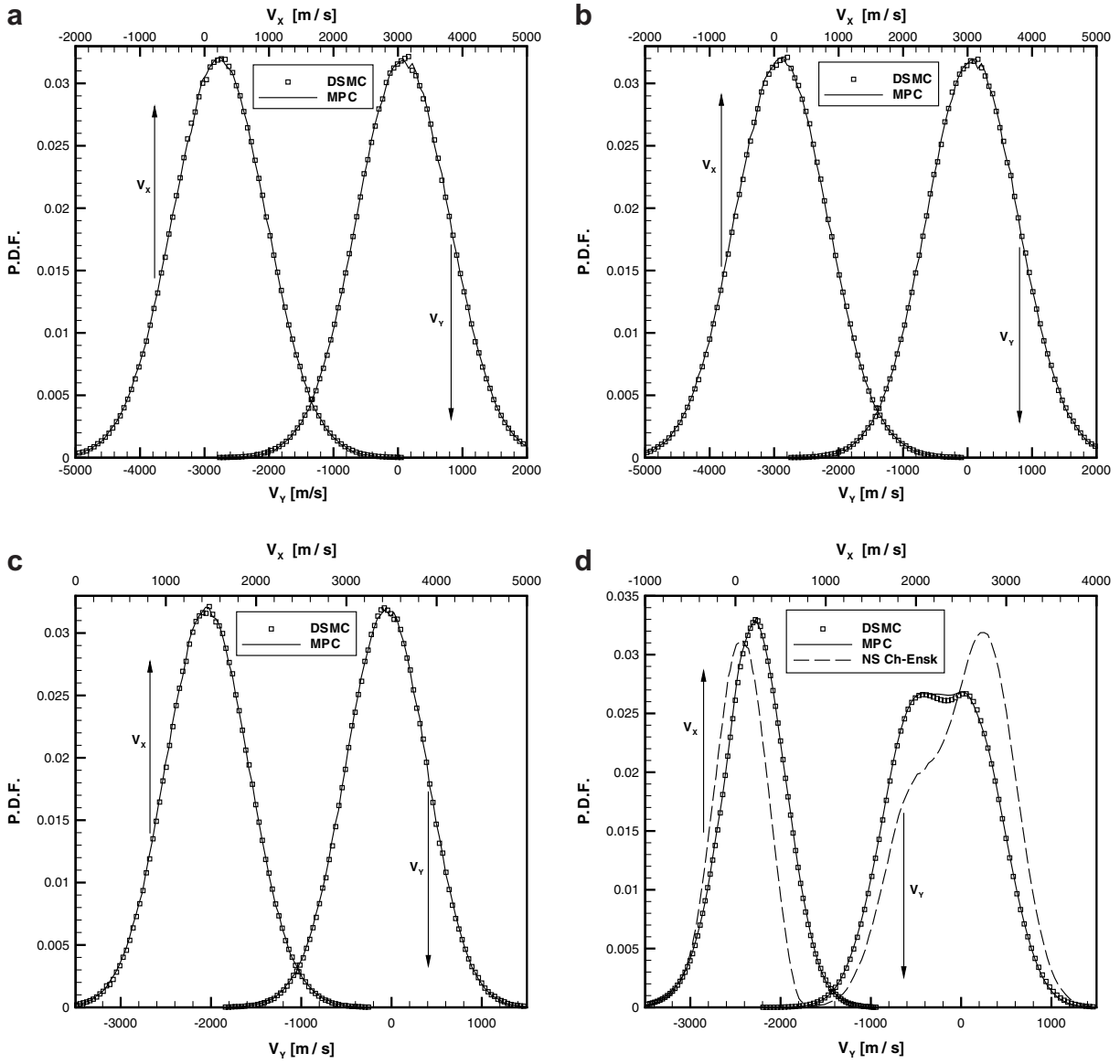

Fig. 8. Local velocity distribution functions for case **M6**. (a) Local vdf at a. (b) Local vdf at b. (c) Local vdf at c. (d) Local vdf at d.

T_{TRA} and T_{ROT}. These non-equilibrium profiles are seen to be significantly different from the single temperature shock profile predicted by the NS equations. As a result of the condition of thermal equilibrium added to the continuum breakdown parameter in Eq. (6), the final post–shock interface is positioned where $T_{\mathrm{TRA}} = T_{\mathrm{ROT}}$. On the other side of the interface, full DSMC results (symbols) show thermal non-equilibrium in the expanding continuum region. Since the rotational energy mode relaxes slower than the translational mode, T_{ROT} remains higher than T_{TRA}. The NS module of the MPC solver assumes a single temperature which can be seen to lie between T_{TRA} and T_{ROT} in the final MPC solution. This assumption of a single temperature is the source of the 3% error in the **M12** simulation. It is also important to note that in the continuum region, the MPC method has shifted the initial NS profile into better agreement with DSMC results. To verify that the error in other flow properties is much less than for T_{TRA}, the density and bulk x-velocity (u) along the streamline are plotted in Fig. 10(b). Here, in both particle and continuum regions, the MPC results are seen to exactly reproduce full DSMC results. Along this streamline (away from the surface), DSMC results only differ

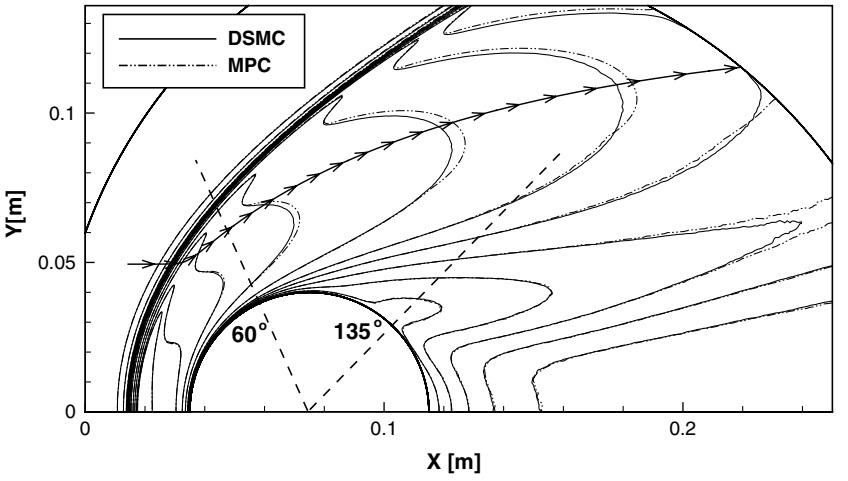

Fig. 9. Contours of translational temperature (case **M12**).

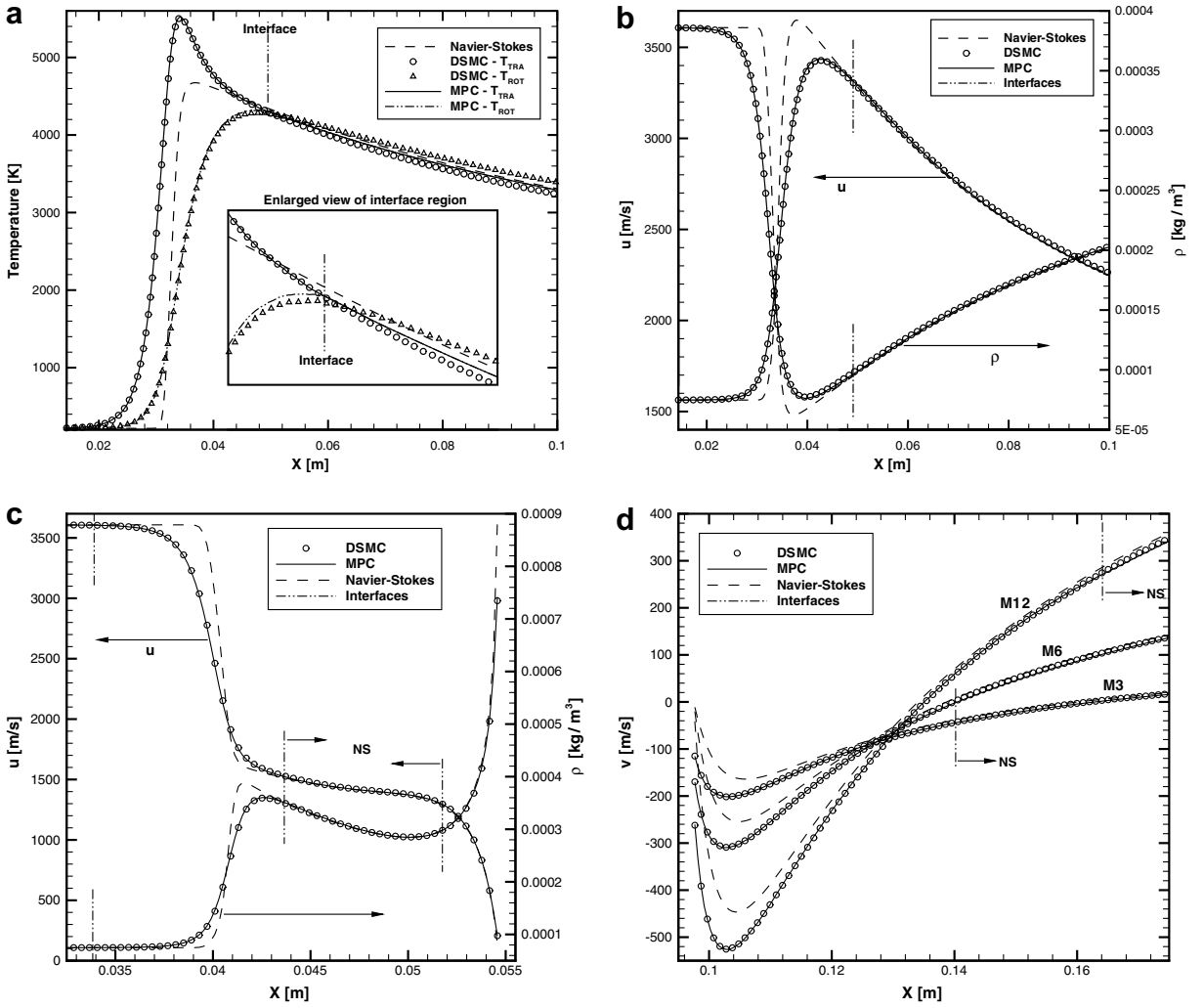

Fig. 10. Validation of macroscopic flow field quantities. (a) Thermal non-equilibrium along a streamline (case **M12**). (b) ρ and u along a streamline (case **M12**). (c) ρ and u along a 60° cut (case **M12**). (d) v along a 135° cut (all cases).

from NS results inside the bow shock. In order to validate the MPC method for the flow close to the cylinder surface, the flow quantities extracted from the 60° and 135° lines shown in Fig. 9 are presented in Fig. 10(c) and (d). In the fore-body flow, Fig. 10(c), the MPC solution is seen to agree exactly with full DSMC results. These profiles of ρ and u differ from the NS profiles inside the shock and also near the cylinder surface. Since the NS solver does not model temperature jump, the NS equations predict a higher density than DSMC at the cylinder surface. Fig. 10(d) shows the significant slip in y-velocity (v) predicted by DSMC at a location 135° around the cylinder surface for all three Mach number flows. The MPC method, which begins with the no-slip NS solution, successfully reproduces these full DSMC results in each case.

6. Computational performance

In limiting particle domains only to regions of highly non-equilibrium flow, the MPC method uses far fewer particles than a full DSMC simulation and as a result saves significant computational time. The efficiency of the MPC method is plotted in Fig. 11 for each simulation. Here the dotted line represents the ideal speedup factor which is simply equal to the ratio of particles in a full DSMC simulation to that required by the MPC method. The time required by a DSMC solver scales nearly linearly with the number of simulated particles. Thus if the MPC method was able to use three times fewer particles and all hybrid operations were free, ideally it could reach the answer 3 times faster. In order to compare the computational time required for DSMC and MPC solutions, simulations are divided into two stages; the time required to reach steady-state and the sampling time. For DSMC simulations, steady-state is reached when the total number of particles stops increasing and becomes constant. For the MPC simulations, as outlined earlier, reaching steady-state involves the time required to reach step (5) of the method (see Section 4.2). A fair comparison then requires both DSMC and MPC simulations to be sampled for the same number of DSMC time steps. Sampling the MPC solution involves fewer particles but also requires coupling to the NS solver and NS iterations of the continuum regions. The total simulation time for both solvers is then the sum of these two stages.

The actual speedup factors for each case are shown in Fig. 11 as the solid symbols. They are seen to be proportional to the number of particles saved by the MPC method (the particle ratio) which is proportional to the amount of the flow field which can be assumed as near-equilibrium flow. The **M12**, **M6** and **M3** cases are seen to run 1.4, 1.6 and 2.2 times faster, respectively. It should be noted that the initial NS solution accounts for roughly 50% of the difference between actual and ideal speedups for each case shown in Fig. 11. This is due to the fact that the NS solver currently uses the same mesh as DSMC, which is refined

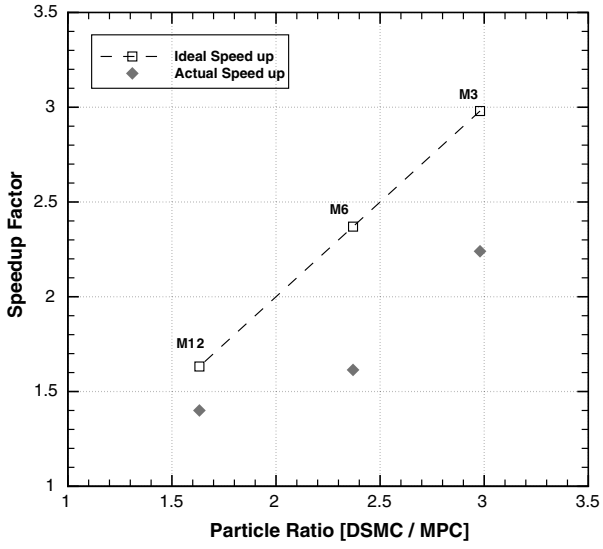

Fig. 11. MPC computational efficiency.

to the mean-free-path. For the cases discussed in this article, a mesh which meets DSMC requirements contains roughly 10 times as many cells as a mesh designed to meet NS requirements. In addition, NS iterations on this fine mesh during the MPC coupling cycle also contribute to the overhead. Thus, by allowing the NS solver to operate on a coarser mesh, a large portion of the MPC overhead could be eliminated. Finally, the third major source of overhead involves the averaging calculations required to track the sub-relaxation average in each DSMC cell at each time-step. Depending on acceleration techniques used in individual DSMC and NS solvers, absolute simulations times may certainly vary, however, speedup factors should apply in general.

7. Summary

In summary, this article details all aspects of a modular particle–continuum (MPC) method which uses the DSMC method in non-equilibrium flow regions and solves the NS equations elsewhere. The MPC method is designed for steady-state hypersonic flows. This allows temporal averaging in conjunction with state-based coupling to effectively control statistical scatter and thus enables smooth information transfer between DSMC and NS regions. The second major aspect of the MPC method is the numerical cycle which ensures that interfaces are positioned in near-equilibrium regions before any information is transferred to the continuum domain. The optimal positioning of interfaces is also enhanced by adding a condition of thermal equilibrium to the definition of continuum breakdown. The MPC method is validated for 2D high-speed flow about a cylinder at a global Knudsen number of 0.01. The MPC method is verified to accurately reproduce full DSMC results for all flow quantities down to the level of the local velocity distribution. Compared with full DSMC simulations, the MPC method requires significantly fewer particles and produces the same results 1.4–2.2 times faster for the cases considered. This performance is achieved using a novel modular implementation which incorporates virtually un-modified, state-of-the-art DSMC and NS solvers into a single hybrid code.

Acknowledgements

This work is sponsored by the Space Vehicle Technology Institute, under NASA grant NCC3-989 with joint sponsorship from the Department of Defense and from the Air Force Office of Scientific Research grant FA9550-05-1-0115. This work is also supported by the Francois-Xavier Bagnoud Foundation.

References

[1] G.A. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Oxford University Press, New York, 1994.
[2] V.I. Kolobov, S.A. Bayyuk, R.R. Arslanbekov, V.V. Aristov, A.A. Frolova, S.A. Zabelok, Construction of a unified continuum/kinetic solver for aerodynamic problems, Journal of Spacecraft and Rockets 42 (4) (2005) 598–606.
[3] V.I. Kolobov, S.A. Bayyuk, R.R. Arslanbekov, V.V. Aristov, A.A. Frolova, S.A. Zabelok, Unified flow solver for aerospace applications, AIAA Paper 06-988, January 2006, Presented at the 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV.
[4] P. Le Tallec, F. Mallinger, Coupling Boltzmann and Navier–Stokes equations by half fluxes, Journal of Computational Physics 136 (1997) 51–67.
[5] L. Pareschi, R.E. Caflisch, An implicit Monte Carlo method for rarefied gas dynamics, Journal of Computational Physics 154 (1999) 90–116.
[6] H.S. Wijesinghe, N.G. Hadjiconstantinou, A discussion of hybrid atomistic–continuum methods for multiscale hydrodynamics, International Journal for Multiscale Computational Engineering 2 (2004).
[7] D.B. Hash, H.A. Hassan, Assessment of schemes for coupling Monte Carlo and Navier–Stokes solution methods, Journal of Thermophysics and Heat Transfer 10 (1996) 242–249.
[8] D.B. Hash, H.A. Hassan, A decoupled DSMC/Navier–Stokes analysis of a transitional flow experiment, AIAA Paper 96-0353, January 1996, Presented at the 34th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV.
[9] H.S. Wijesinghe, R.D. Hornung, A.L. Garcia, N.G. Hadjiconstantinou, Three-dimensional hybrid continuum–atomistic simulations for multiscale hydrodynamics, Journal of Fluids Engineering 126 (2004) 768–777.
[10] N.G. Hadjiconstantinou, A.L. Garcia, M.Z. Bazant, G. He, Statistical error in particle simulations of hydrodynamic phenomena, Journal of Computational Physics 187 (2003) 274–297.
[11] D.B. Hash, H.A. Hassan, Two-dimensional coupling issues of hybrid DSMC/Navier–Stokes Solvers, AIAA Paper 97-2507, June 1997, Presented at the 32nd AIAA Thermophysics Conference, Atlanta, GA.
[12] R. Roveda, D.B. Goldstein, P.L. Varghese, Hybrid Euler/particle approach for continuum/rarefied flows, Journal of Spacecraft and Rockets 35 (3) (1998) 258–265.

[13] R. Roveda, D.B. Goldstein, P.L. Varghese, Hybrid euler/direct simulation Monte Carlo calculation of unsteady slit flow, Journal of Spacecraft and Rockets 37 (6) (2000) 753–760.
[14] T.E. Schwartzentruber, I.D. Boyd, A hybrid particle–continuum method applied to shock waves, Journal of Computational Physics 215 (2) (2006) 402–416.
[15] S. Dietrich, I.D. Boyd, Scalar and parallel optimized implementation of the direct simulation Monte Carlo method, Journal of Computational Physics 126 (1996) 328–342.
[16] I.D. Boyd, Analysis of rotational nonequilibrium in standing shock waves of nitrogen, AIAA Journal 28 (1990) 1997–1999.
[17] L.C. Scalabrin, I.D. Boyd, Development of an unstructured Navier–Stokes solver for hypersonic nonequilibrium aerothermodynamics, AIAA Paper 05-5203, June 2005, Presented at the 38th AIAA Thermophysics Conference, Toronto, Ont., Canada.
[18] R.W. MacCormack, G.V. Candler, The solution of the Navier–Stokes equations using gauss-seidel line relaxation, Computers and Fluids 17 (1989) 135–150.
[19] I.D. Boyd, G. Chen, G.V. Candler, Predicting failure of the continuum fluid equations in transitional hypersonic flows, Physics of Fluids 7 (1) (1995) 210–219.
[20] W.-L. Wang, I.D. Boyd, Predicting continuum breakdown in hypersonic viscous flows, Physics of Fluids 15 (2003) 91–100.
[21] A.L. Garcia, B.J. Alder, Generation of the Chapman–Enskog distribution, Journal of Computational Physics 140 (1998) 66–70.
[22] Q. Sun, I.D. Boyd, Evaluation of macroscopic properties in the direct simulation Monte Carlo method, Journal of Thermophysics and Heat Transfer 19 (3) (2005) 329–335.