Computers & Fluids 39 (2010) 1136–1145

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

Implementation of a transient adaptive sub-cell module for the parallel-DSMC code using unstructured grids

C.C. Su [a], K.C. Tseng [b], H.M. Cave [a,c], J.S. Wu [a,*], Y.Y. Lian [b], T.C. Kuo [b], M.C. Jermy [c]

[a] Department of Mechanical Engineering, National Chiao Tung University, 1001 Ta-Hsueh Road, Hsinchu 30050, Taiwan
[b] National Space Organization, 8F, 9 Zhan-Ye 1st Road, Hsinchu Science Park, Hsinchu, Taiwan
[c] Department of Mechanical Engineering, University of Canterbury, Private Bag 4800, Christchurch 8140, New Zealand

ARTICLE INFO

Article history:
Received 10 January 2009
Received in revised form 5 February 2010
Accepted 17 February 2010
Available online 1 March 2010

Keywords:
DSMC
Sub-cell
Unsteady
Vortex shedding

ABSTRACT

A method of transient adaptive sub-cells (TAS) suitable for unstructured grids that is modified from the existing one for the structured grids of DSMC is introduced. The TAS algorithm is implemented within the framework of a parallelized DSMC code (PDSC). Benchmarking tests are conducted for steady driven cavity flow, steady hypersonic flow over a two-dimensional cylinder, steady hypersonic flow over a cylinder/flare and the unsteady vortex shedding behind a two-dimensional cylinder. The use of TAS enables a reduction in the computational expense of the simulation since larger sampling cells and less simulation particles can be employed. Furthermore, the collision quality of the simulation is maintained or improved and the preservation of property gradients and vorticity at the scale of the sub-cells enables correct unsteady vortex shedding frequencies to be predicted. The use of TAS in a parallel-DSMC code allows simulations of unsteady processes at a level to be carried out efficiently, accurately and with acceptable computational time.

1. Introduction

The Direct Simulation Monte Carlo (DSMC) method is a computational tool for the simulation of flows in which effects at the molecular scale become significant [1]. The Boltzmann equation, which is appropriate for modelling these rarefied flows, is extremely difficult to solve numerically due to its high dimensionality and the complexity of the collision term. Simplification of the collision term in the equation by the Bhatnagar–Gross–Krook (BGK) technique [2] has spawned a number of numerical solvers, such as the model Boltzmann equation (MBE) solver developed by Yang and Huang [3]; however these methods are still in their infancy as practical simulation tools.

DSMC provides a particle-based alternative for obtaining realistic numerical solutions in which the movement and collision behaviour of a large number of representative particles within the flow field are decoupled over a time-step which is a small fraction of the local mean collision time. The computational domain itself is divided into either a structured or unstructured grid of cells which are then used to select particles for collisions on a probabilistic basis and also are used for sampling the macroscopic flow properties. Intermolecular collisions are handled probabilistically using phenomenological models which designed to reproduce real fluid behaviour when the flow is examined at the macroscopic level. These models vary in their sophistication; however the models used in most applications include the variable hard sphere (VHS) [4] and the variable soft sphere (VSS) [5] models. DSMC has been shown by Nanbu [6] and Wagner [7] to provide a solution to the Boltzmann equation as the number of simulated particles tends toward the true value within the flow field.

The incorporation of sophisticated features in the DSMC algorithm, such as nearest-neighbour collisions, adaptive cell structures and variable time-step (VTS) schemes, and the continued increase in computing power available to the general user, has greatly increased the ability of the method to model a wide variety of applications including hypersonic flows, space applications, micro electro mechanical system (MEMS) devices and deposition processes, among others. A recent study by Gallis et al. [8] demonstrates that these 'sophisticated procedures' can greatly increase the accuracy and computational efficiency of the basic DSMC algorithm on structured grids. Despite these advances, the DSMC scheme becomes increasingly computationally and memory intensive as the flow approaches continuum limit and when unsteady flows are modelled, which has tended to limit the method when single computer processors are employed. The development of parallel computer processing, whereby the computational load is spread over a number of computers, represents an opportunity to simulate near-continuum and unsteady flows with acceptable run-times and a number of sophisticated parallel-DSMC schemes

* Corresponding author. Tel.: +886 3 573 1693; fax: +886 3 611 0023.
E-mail address: chongsin@faculty.nctu.edu.tw (J.S. Wu).

0045-7930/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2010.02.003

have been implemented and reported in the literature. Several recent implementations of parallel-DSMC include those Dietrich and Boyd [9], Ivanov et al. [10], LeBeau [11] and Wu and Lian [12]. The parallel-DSMC Code (PDSC) developed by Wu's group has been successfully used to model a number of flows including flow through a drag pump [13], hypersonic flow past a cylinder [14] and under-expanded jet flow [15]. PDSC will be discussed in greater detail in Section 2.1.

As mentioned, the simulation of a near-continuum gas flow results in an increase in the computational expense and memory requirements of the simulations because finer cells and more simulated particles are required. This is because traditionally the dimensions of the sampling cells are set to be equal to approximately 1/3–1/2 times the local mean free path, so increased density results in a greater number of cells and more simulated particles to ensure an adequate statistical sample within these cells. Sampling cells of this size ensure that the distances between particles selected for collisions are maintained at below one mean free path. A measure of the "collision quality" of a simulation is the ratio of the mean spacing of particles selected for collision (mean collision spacing) to the local mean free path (mcs/mfp). The mcs/mfp ratio should be much less than unity for a simulation to be considered accurate [8].

The use of a large number of properly resolved sampling cells will incur a high computational expense for simulating unsteady processes like Pulsed Pressure Chemical Vapour Deposition (PPCVD) which have regions of relatively high density and large gradients in macroscopic properties [16,17]. A potential method of overcoming this problem, whilst maintaining a high collision quality, is to divide the sampling cells into sub-cells for the selection of collision partners. This means that the distance between collision partners is maintained at the dimensions of the sub-cells, whilst the resolution of the macroscopic properties is at the dimensions of the sampling cells. Gallis et al. [8] have demonstrated that the use of sub-cells or virtual sub-cells which result in nearest-neighbour collisions can lead to, for a one-dimensional simulation, a coarsening of the grid by approximately eight times whilst still maintaining the same simulation accuracy.

The goal of this paper is to present the implementation of a transient adaptive sub-cell routine for unstructured grids into the parallel-DSMC (PDSC) code developed by Wu's group [12–15]. The method is validated by simulating steady driven cavity flow, steady Mach 10 flow over a two-dimensional cylinder, steady Mach 12.4 flow over a hollow cylinder/flare and unsteady vortex shedding after a two-dimensional cylinder body.

2. Numerical method

2.1. Parallel implementation of DSMC

In the DSMC algorithm, particle movement and collision events are treated independently and occur locally, making the code is highly suitable for parallelization. This can be achieved through decomposition of the physical domain into groups of cells which are then distributed among the parallel processors. Each processor executes the DSMC algorithm in serial for all particles and cells in its own domain. Parallel communication between processors is required when particles cross the domain boundaries requiring particles to be transferred between processors. To achieve high parallel efficiency it is necessary to minimize the communication between processors while maintaining a balance between the computational load on each processor, which is best achieved adaptively during the run-time. In the present study, we have adapted the previously developed parallel-DSMC Code (PDSC) which has been described in detail in the literature [12–15] and will only be outlined briefly here.

PDSC enables simulation of two-dimensional, axisymmetric or three-dimensional problems. The algorithm is implemented on an unstructured mesh using a particle ray-tracing technique, which takes advantage of the cell connectivity information provided by the mesh data and is able to handle complex boundary geometry. PDSC utilizes the multi-level graph partitioning tool ParMETIS to decompose the computational domain and distribute the cells among the processors. A stop-at-rise (SAR) algorithm is used to determine when to dynamically repartition and re-distribute the computational load between processors based on the value of a degradation function which compares the computational cost of repartition to the idle time for each processor. The transfer of particle data between the processors only occurs when particles strike the inter-processor boundaries and after all other particles on each processor have been moved, thus minimizing communication between processors and maximizing the parallel speed-up. During calculation, the mesh can be iteratively refined using the h-refinement technique whereby local grid points are added to improve the cell distribution according to the solution based on some adaptation criteria (for example, flow field density or local Knudsen number). A simple cell-quality control method is used to maintain the integrity of the mesh during this process.

Other special features have been developed to enhance the computational efficiency, flexibility and utility of PDSC. These include pressure boundary treatment [18], spatial and temporal variable time-step schemes [13], the implementation of a conservative weighting scheme to efficiently deal with gas flows with trace species [19], the implementation of an unsteady sampling algorithm [20] and a gas phase chemistry module for simulating chemical reactions in hypersonic air flows [21].

2.2. Transient sub-cells method

As mentioned previously, the implementation of sub-cells in DSMC allows the maintenance of good collision quality within the simulation, even for grids which are "under-resolved" (that is, if the sampling cells are bigger than the recommended setting of 1/3–1/2 times the local mean free path). Running simulations with under-resolved sampling cells which employ sub-cells results in a reduction in the computational and memory requirements of the simulation, albeit at the cost of a reduction in the possible sampling resolution of the macroscopic properties, but without sacrificing simulation accuracy.

The methods used to for the implementation of sub-cells within the literature vary somewhat. Older versions of Bird's code [1] employs a fixed number of sub-cells per sampling cell, however the latest version of DS2V code generates a transient grid to each cell at a time during the collision routine such that there is approximately one particle per sub-cell [8]. These sub-cell procedures do not guarantee nearest-neighbour collisions, because they do not determine the distance between particles explicitly; however they ensure particles which are selected for collision are closely spaced. The use of virtual sub-cells, whereby the distances between the particle selected for collision and all other particles in the cell are simply explicitly calculated and the nearest particle chosen, was introduced in NASA's DAC code [22] and has since been included in DS2V.

It is important to note that the codes mentioned above utilize structured grids. In PDSC unstructured grids are used, requiring an adaptation of the transient sub-cells scheme whilst maintaining the simplicity of the scheme and its low computational cost. In PDSC, the sampling cells are divided into sub-cells during the collision routine. Because the sub-cells only exist in one sampling cell at a time, and only during the collision routine, they can be consid-

ered "transient sub-cells" which will have negligible computer memory overhead. In a two-dimensional flow, for example, these sub-cells are quadrilateral which reduces the complexity of sub-dividing the sampling cell and greatly facilitates particle indexing. The size of the sub-cells is indirectly controlled by the user, who inputs the desired number of particles per sub-cell, P. The program then determines the dimensions of the sub-cell array based on the number of particles within the cell, N_{parts}. For example, for a two-dimensional grid, the number of sub-cells in each coordinate direction is $\sqrt{N_{parts}/P}$, rounded to the nearest integer.

Fig. 1 shows the way in which unstructured sampling cells are divided into sub-cells. As can be seen there may be sub-cells which are entirely outside the boundary of the sampling cell; however this has no affect on the collision routine. The concept can be easily extended to three-dimensional sampling cells. Note that using the $\sqrt{N_{parts}/P}$ as a guide for the deciding number of sub-cells is not perfect for a high-aspect ratio background cell. For example, consider two triangular background cells with the same area as shown in Fig. 2. Obviously, it needs a larger domain to cover a "skewed" triangular background cell (Fig. 2b) than a "regular" triangular (Fig. 2a). Suppose both of these cells have the same simulation particles (e.g., 50) and the total number of sub-cells is the same based on the estimation as proposed in the paper ($\sqrt{N_{parts}/P}$). If the particles are randomly distributed within the background cell, then the "skewed" triangular background cell will have more simulation particles. This becomes even worse when the aspect ratio of the triangular cell increases. However, appearance of triangular cells having high-aspect ratio could be reasonably prevented during the process of mesh generation. We have prepared a simple code to check the generated mesh if the aspect ratio of any cell exceeds some acceptable value (e.g., 2.5). Or we can use smaller value of P (=1–2) to obtain more accurate data, which will be shown later. Smaller P tends to provide more accurate simulation but higher computational cost since more sub-cells are needed. This is a trade-off between accuracy and computational time that a user has to keep in mind. In the present paper, we have employed $P = 2$ throughout the study.

During the collision routine, the collision procedure is processed cell by cell. The first particle is chosen at random from some point within the whole sampling cell. The sub-cell in which the particle lies is then determined and if other particles are in the same sub-cell then the second collision partner is selected from these particles. If the first particle is alone within the sub-cell, then adjacent sub-cells are scanned for a collision partner. These sub-cell routines ensure the collision partners are closely spaced, even within under-resolved sampling cells, with a minimal computational and memory overhead.

Particles are also prevented from colliding again their last collision partner [23]. The basis of this is that collision between particles which just collided with each other is unphysical, since the particles must be moving away from each other after the first collision. A minor modification was made to PDSC to prevent particles colliding with their last collision partner. This involved the creation of an array in which the last collision partner for every particle is stored and if the two particles are subsequently chosen for collision without having collided with any other particle, the collision is rejected.

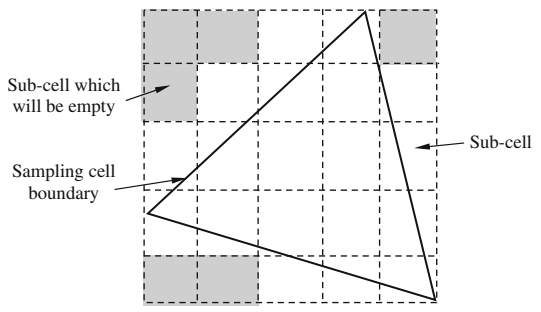

Fig. 1. Schematic showing the division of an unstructured sampling cell into sub-cells using the transient adaptive sub-cell (TAS) method.

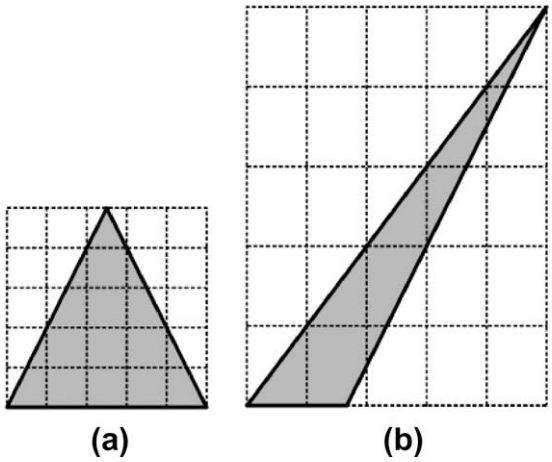

Fig. 2. Average particle number per sub-cell is (a) 3.125 and (b) 6.25 for 50 particles in the background triangular cell.

3. Code validation

3.1. Steady driven cavity flow

The first validation study involves the simulation of a square driven cavity flow. Here the simulation domain is bounded on all sides by diffusely reflecting walls at a temperature of 300 K. All walls are stationary except the upper wall which moves at Mach 1.1 (354.2 m/s). The cavity is initially filled with stationary argon gas at 300 K with a Knudsen number based on the cavity dimensions Kn = 0.01, the upper plate is impulsively started and the flow allowed to reach steady state after which 15,000 time-steps are sampled every second time-step. Three simulation runs were conducted as detailed in Table 1. In each case the simulations were conducted on a ten 2.2 GHz AMD processor cluster.

Fig. 3a shows the contours of total temperature for each of the cases, while Fig. 3b shows the temperature profile through the vertical centreline of the simulation domain. Here cases I and III show essentially the same results, while case II shows a different result demonstrating that the use of TAS enables replication of the benchmark results with considerably less computational cost (about 1/16 times in this case).

Fig. 4 shows contours of the mcs/mfp ratio for each of the cases. In the benchmark case (case I), the ratio is maintained below unity. Note the maximum value was found to be 0.88 near the upper right-hand corner of the domain, although it is almost invisible in the plot since the region where it is located is very small. Where the sampling cells are under-resolved and TAS is not used (case II) the mcs/mfp ratio is well above unity for most flow regions (maximum value: 2.60), indicating that collision partners are frequently spaced at over one mean free path apart which will result in simulation inaccuracies. In case III, where the sampling cells are under-resolved but TAS is employed, the ratio is maintained at well below unity in all regions of the flow (maximum value: 0.72). In fact, the collision quality in case III is roughly the same as the benchmark simulation (case I). This demonstrates that the use of

Table 1
Simulation cases for the steady two-dimensional driven cavity flow.

Case	TAS function	Cell no.	Cell size	Particles per cell	Total particles
I (Benchmark)	OFF	400×400	$\Delta x \sim 1/4\lambda$	25	4,000,000
II (Non-TAS)	OFF	100×100	$\Delta x \sim \lambda$	25	250,000
III (TAS)	ON	100×100	$\Delta x \sim \lambda$	25	250,000

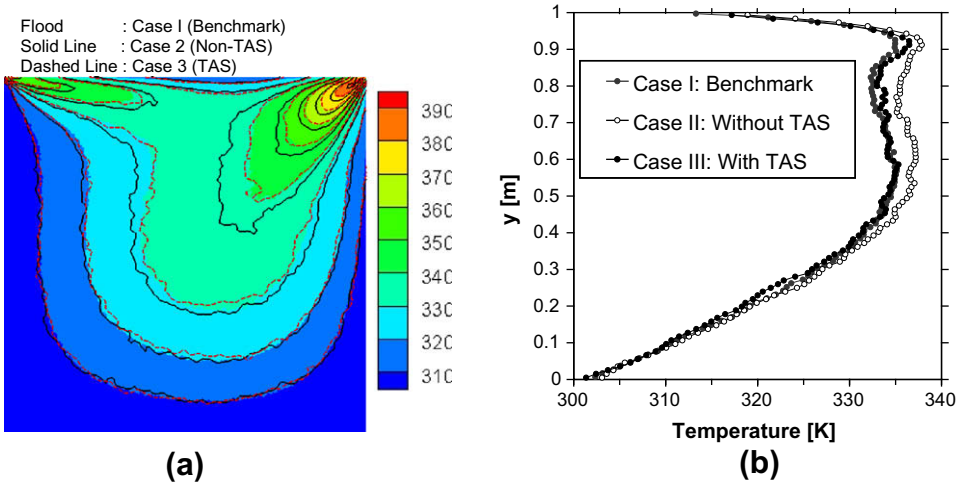

Fig. 3. (a) Contours of total temperature (K) and (b) the temperature profile through the vertical centreline of the domain for driven cavity flow (Mach 1.1, Kn = 0.01) showing the effect of employing transient adaptive sub-cells (TAS).

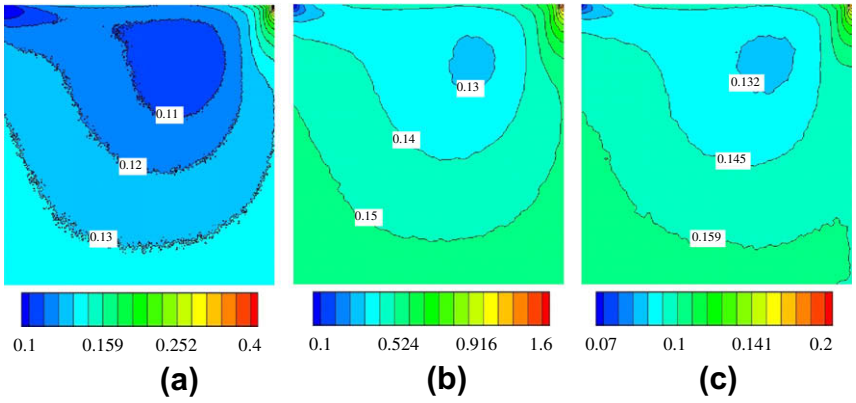

Fig. 4. Contours of mean collision spacing to mean free path (mcs/mfp) ratio as a measure of collision quality for driven cavity flow (Mach 1.1, Kn = 0.01) for (a) the benchmark case (case I), (b) the case not employing TAS (case II) and (c) using TAS (case III). An exponential scale is used in every case.

TAS enables accurate simulations to be carried out with under-resolved sampling cells.

The total simulation time for these three cases were 34,107, 2017 and 2303 CPU-seconds respectively, demonstrating that TAS introduces a computational overhead of about 15% but is able to replicate the results of the benchmark case using 1/16th of the number of simulation particles and sampling cells and about 7% of the computational expense. This shows the TAS makes the simulation becomes more efficient.

In addition, a sensitivity study of the desired number of simulators per sub-cell (P) is also proposed here for reference. Fig. 5 shows the temperature profiles of varying P (=1, 2, 6). All the cases with TAS perform much better than that without using TAS. It also can be seen that the simulation results with a smaller value of P (=1 and 2) are in better agreement with the benchmark data than a larger value of P (=6). However, the computational time also increases

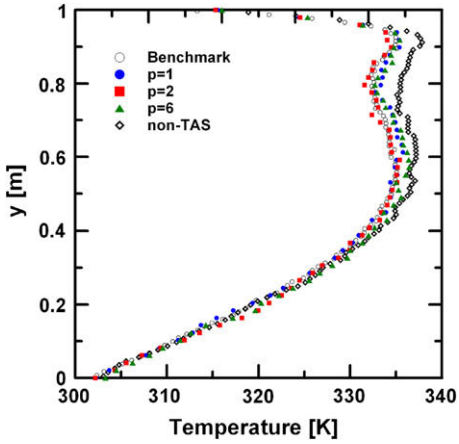

Fig. 5. The sensitivity study of varying P of using transient adaptive sub-cells (TAS).

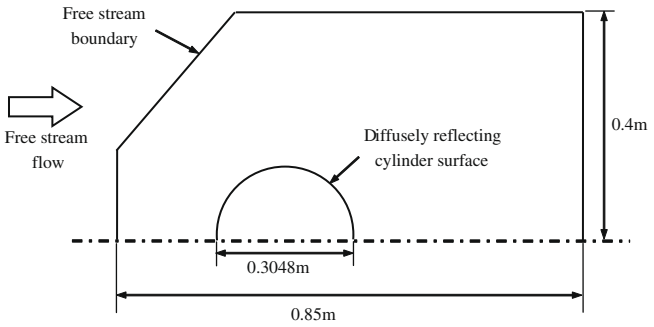

Fig. 6. Geometry used in simulation of hypersonic flow over a two-dimensional cylinder (not to scale. Note that only primary dimensions are shown for clarity).

slightly with decreasing value of P, which are 3.02, 2.63, and 2.35 h for the cases using $P = 1$, 2, and, 6, respectively. Therefore, users can choose a suitable valve of P based on their computational resource and requirement.

3.2. Steady hypersonic flow over a two-dimensional cylinder

A benchmark test for DSMC code recently adopted by Bird [24] is the Mach 10 hypersonic flow of argon at 200 K over a 12 in. circular cylinder with a thermally diffuse surface at 500 K and a nominal free-stream Knudsen number of 0.01, as shown in Fig. 6. This case was first used by Lofthouse, Boyd and Wright [25] as a comparison between a Navier–Stokes solver and the MONACO DSMC code.

Table 2
Comparison of the surface properties for Mach 10 hypersonic flow over a circular cylinder predicted by various DSMC solvers.

Solver	Peak heat flux (W/m^2)	Total drag (N)
PDSC (without TAS)	44,813	41.90
PDSC (with TAS)	40,888	39.16
DS2V	38,400	39.76
DAC	38,500	39.71
SMILE	39,000	39.76
MONACO	39,319	40.00

Bird has compared the results for the cylinder drag and peak surface heat transfer for his own single processor DS2 V code and a number of other DSMC codes including Boyd's MONACO, Ivanov's SMILE, NASA's DAC and his own obsolete DS2G code. Using the results, he demonstrated that the use of sophisticated DSMC procedures such as nearest-neighbour collisions, cell adaptation and variable time-steps, can produce accurate engineering results with a fraction of the computational time of DSMC codes not employing these procedures.

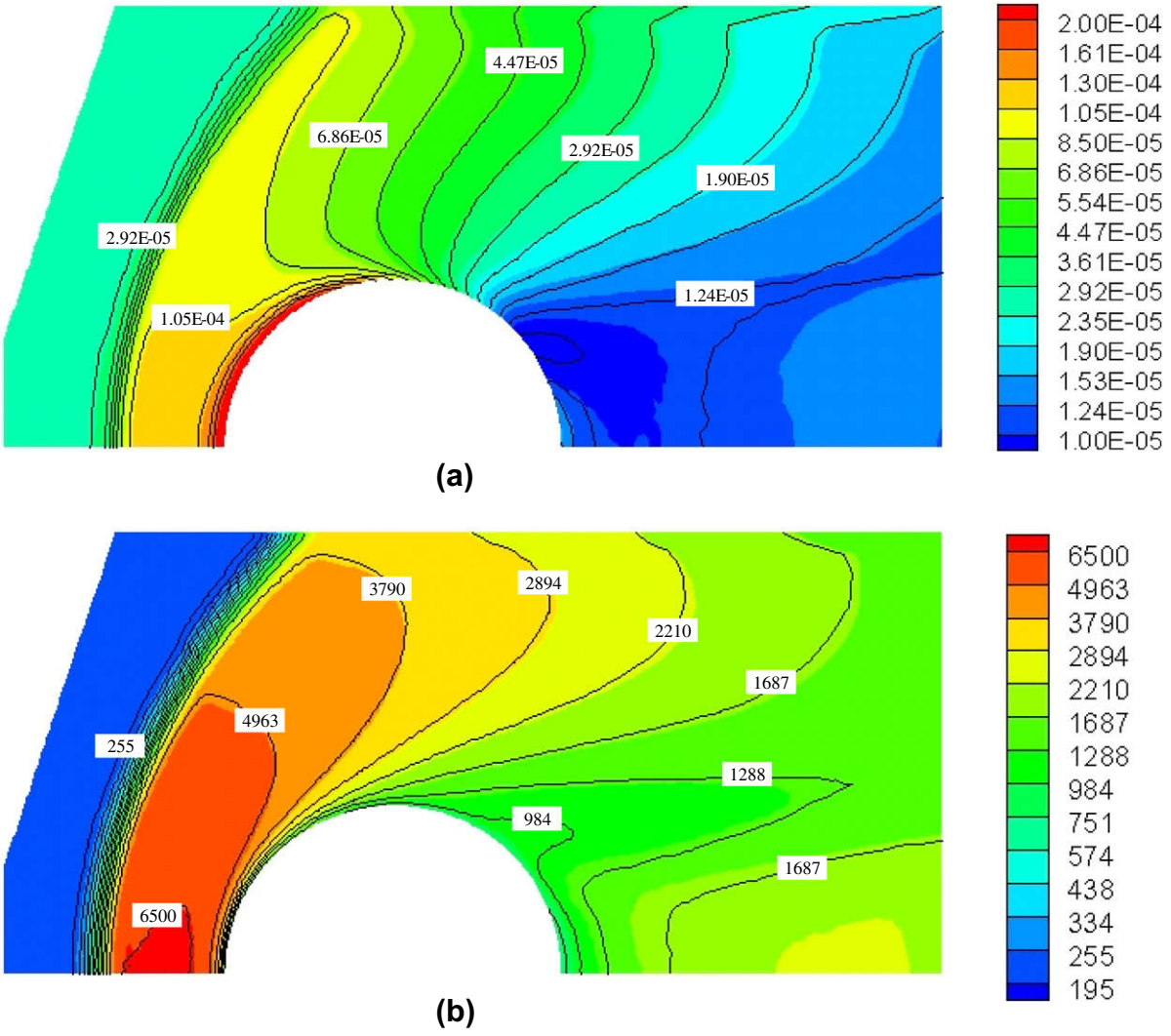

Fig. 7. Contours of (a) density (kg/m^3) and (b) total temperature (K) for the Mach 10 flow over a two-dimensional cylinder with (flood) and without transient adaptive subcells (line). The scale is exponential.

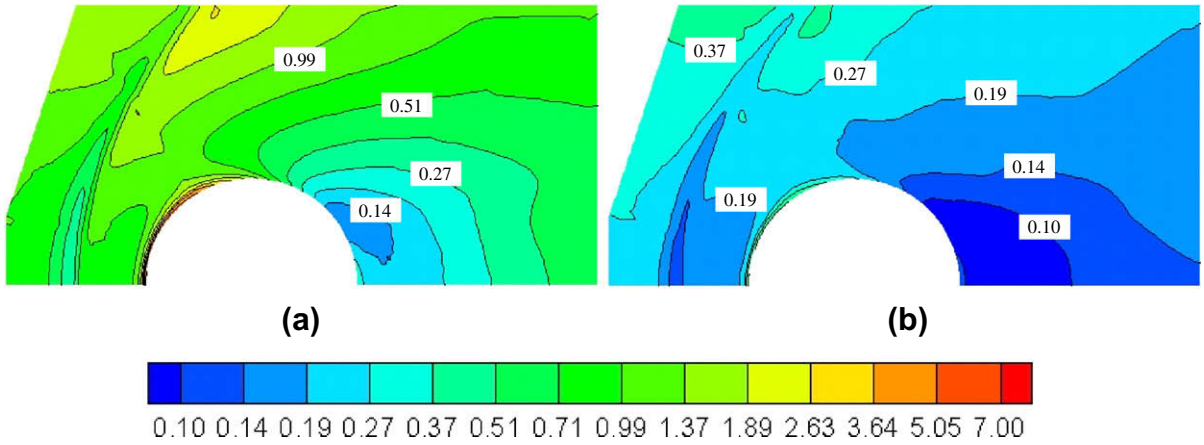

Fig. 8. Contours of collision quality (mean collision spacing to local mean free path ratio) for Mach 10 hypersonic flow over a cylinder (a) without transient adaptive sub-cells (TAS) and (b) with TAS. The scale is exponential.

In order to benchmark PDSC against the other contemporary DSMC codes and to assess the effect of TAS, PDSC was used to simulate the case. The number of cells in the simulation is 9900, with 150 surface elements on the cylinder wall, and the total particle number peaks at 88,000. The computational times for the simulation with and without TAS are 4371 and 3102 s, respectively. Table 2 shows the peak heat fluxes and drag coefficients predicted by the solvers listed above along with PDSC with and without TAS. The PDSC result using TAS shows a close agreement to the other solvers; however none of these solutions can be considered a benchmark solution as the way in which the surface properties are sampled by the other solvers is unclear. The MONACO simulation, for example, is the most expensive computationally as it employs the largest number of simulation particles and sampling cells and, therefore could be assumed to be the most accurate simulation; however as the surface sampling procedures are unknown, this is not a strong assumption. As has been pointed out by Cave [17], Bird's results use 2° surface sampling intervals which effectively smooth the properties at the surface and thus reduce the peak heat flux value, whereas PDSC samples the surface properties at the same resolution as the flow field sampling cells. Where a 2° surface sampling method applied to the PDSC sample (with TAS active) the peak heat flux reduces to 39,385 W/m^2.

Fig. 7 shows contours of density and total temperature for the flow over the cylinder with and without TAS, while Fig. 8 shows a comparison of the collision quality between runs conducted with and without TAS (other conditions such as the number and size of the sampling cells remain the same). From Fig. 8 it can be seen that employing TAS greatly increases the quality of the collision and consequently it should increase the accuracy of the simulation results, at the cost of an increase in computational expense (approximately 40% in this case).

3.3. Steady hypersonic flow over a cylinder/flare

Hypersonic flow over a 30°-hollow cylinder/flare, as shown in Fig. 9, has been investigated experimentally by Holden [26] and

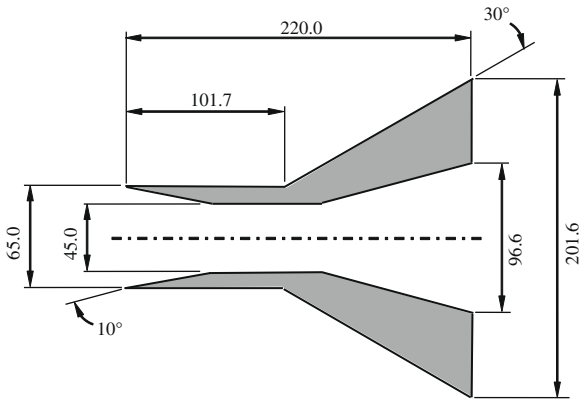

Fig. 9. Geometry used in the simulation of hypersonic flow over a hollow cylinder/flare (dimensions in mm).

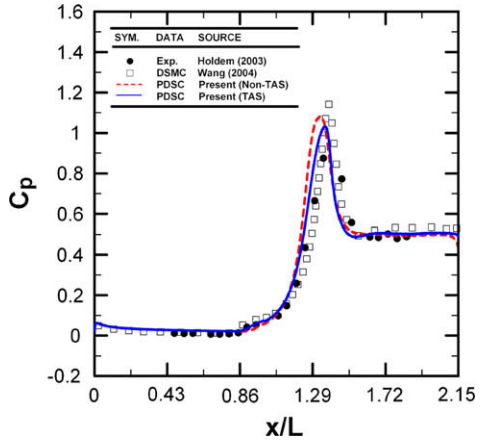

Fig. 10. Pressure coefficient along a hollow cylinder/flare surface.

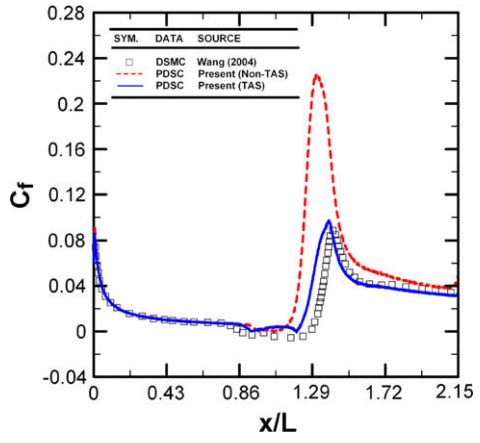

Fig. 11. Skin friction coefficient along a hollow cylinder/flare surface.

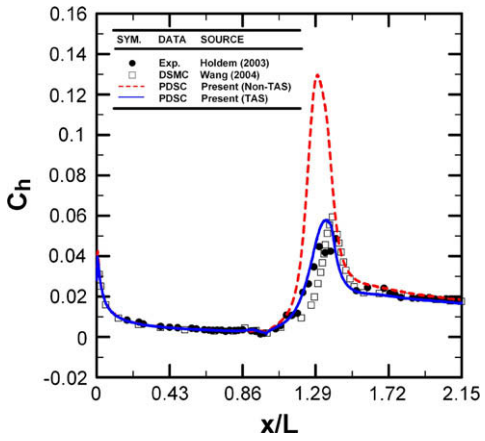

Fig. 12. Heat transfer coefficient along a hollow cylinder/flare surface.

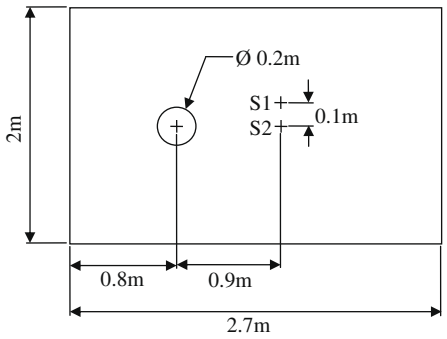

Fig. 13. Geometry used in simulation of unsteady vortex shedding behind a two-dimensional cylinder including the sampling points for the components of *x*-velocity (S1) and *y*-velocity (S2). All outer boundaries correspond to free-stream flow conditions.

numerically by Wang [27]. As a further test of transient adaptive sub-cells, PDSC was used to simulate the flow and compare with results of simulation and ground test. Simulation conditions are listed as follows. Variable hard sphere nitrogen was used with a free-stream Mach number $Ma_\infty = 12.4$, density $\rho_\infty = 5.566\text{E}{-4}$ kg/m^3 and temperature $T_\infty = 95.6$ K. The resulting Knudsen and Reynolds numbers based on the length of the cylinder of 101.7 mm are 8.264×10^{-4} and 6.35×10^4, respectively. The surface of the cylinder and the flare are modelled as fully thermally accommodating diffuse walls at a temperature flat $T_w = 297.2$ K. A constant rotational energy exchange model is used with a rotational collision number of $Zr = 5$. Vibrational energy transfer is neglected due to the low flow temperatures involved.

In Wang's simulations [27], the well-known MONACO DSMC code were used to simulate the hollow cylinder/flare and the double cone cases. The cell size he used for was also too coarse for the flow, especially in the area near the oblique shock. In order to obtain better collision behaviours, a sub-cell scheme with more simulated particles was utilized to ensure the reasonable collision

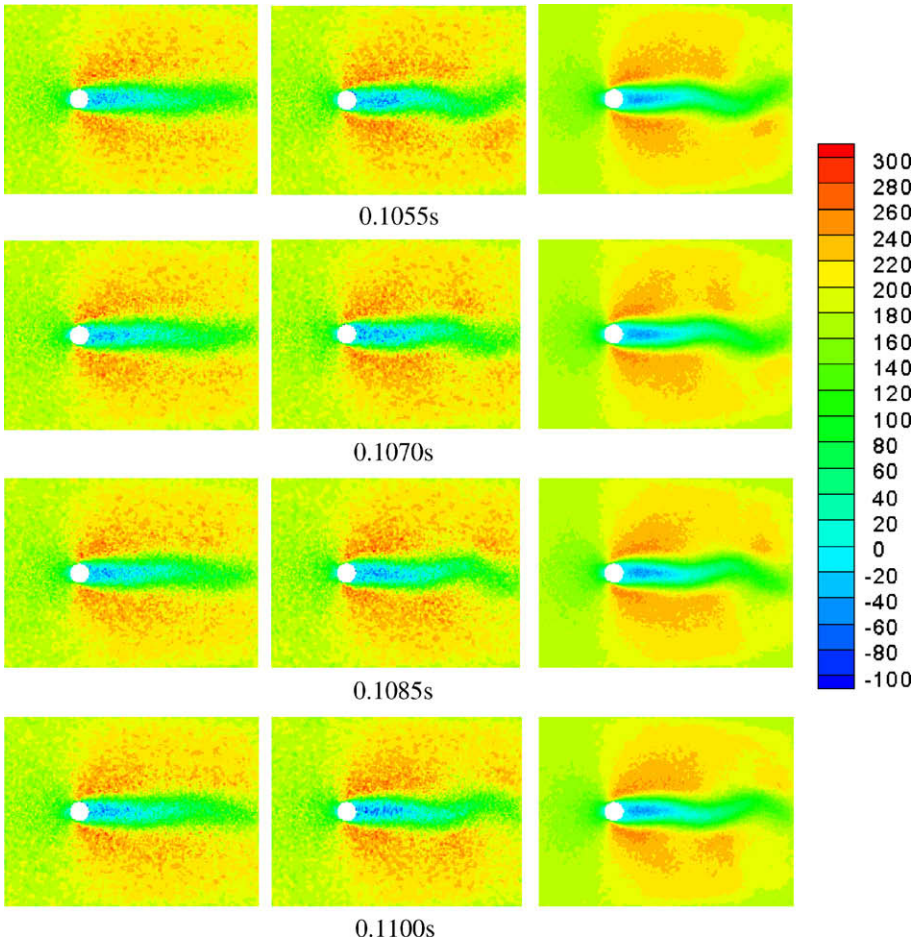

Fig. 14. Contours of stream-wise velocity (m/s) at different times for the vortex shedding behind a cylinder without TAS (left) with TAS (centre) and after processing with DREAM (right).

pairs selection. In the current simulation, the total cell number is about 200,600 which is similar to that used in the simulations by Wang, and 23 million simulated particles are used (compared to 27 million by Wang). Two axisymmetric PDSC simulations were conducted, with and without the TAS module. Both simulations used 20 CPUs with the computational time for the TAS simulation being 40 h, and the non-TAS simulation being 34 h.

Figs. 10–12 show the aerodynamic coefficients along the cylinder and flare surfaces from the two simulations. The experimental data from Holden [26] and numerical data from Wang [27] are also included for comparison except the experimental skin friction coefficient is unavailable. In these figures, the hollow square and solid circle are Wang's DSMC simulation data and Holden's experimental results. Fig. 10 illustrates the pressure coefficient and it shows both the PDSC simulations with and without TAS can obtain satisfied results by comparing with references. However, there exists significant difference between simulations with and without using TAS in skin friction and heat transfer coefficients, which are shown as Figs. 11 and 12, respectively. As it can be seen, the simulation with TAS does improve the simulation results even when the cell size does not meet the DSMC cell criterion.

4. Vortex shedding simulations

Vortex formation and shedding behind bluff bodies is a physical phenomenon which has been of interest to many researchers over the last 150 years, during which time there have been have been extensive theoretical, experimental and numerical investigations. The vast majority of these studies have been made on liquids or gases at continuum levels of density, where effects at the molecular scale are insignificant. For rarefied gases, vortex formation and shedding remains an interesting phenomenon; however there are limited studies in the literature. Meiburg [28] initially investigated the problem numerically using molecular dynamics and Direct Simulation Monte Carlo (DSMC) simulations; however, DSMC failed to resolve the vortex structure possibly due to the collision cells being far too large. The problem was again investigated by Koura [29], Bird [30] and Talbot-Stern and Auld [31] for flow over a flat plate normal to the flow direction, who all found that DSMC was capable of generating the large-scale unsteady vortices. Talbot-Stern and Auld [31] noted a significant deviation in the shedding frequency to that observed experimentally by Roshko [32] in the continuum limit and thoroughly investigated parametrically the effect of their simulation conditions on this frequency. Limited information on collision quality was provided in these papers.

In this study, sub-cells were employed to investigate their effect on vortex formation and shedding frequency behind a two-dimensional cylinder of diameter D in subsonic (Mach 0.6) flow. The Knudsen number based on free-stream conditions was 0.00833, the flow and cylinder surface temperature was 300 K and flow Reynolds number was 80. The unsteady DSMC sampling technique developed in Cave et al. [20] was employed to simulate this flow using an unstructured triangular grid with 49,104 elements and a time-step of 5×10^{-8} s. The flow was sampled using 100 time-averaged steps at each temporal point of interest. Fig. 13 shows the simulation geometry and the point behind the cylinder where the sampling of the local components of flow velocity was carried out (4.5 D downstream of the centre of the cylinder).

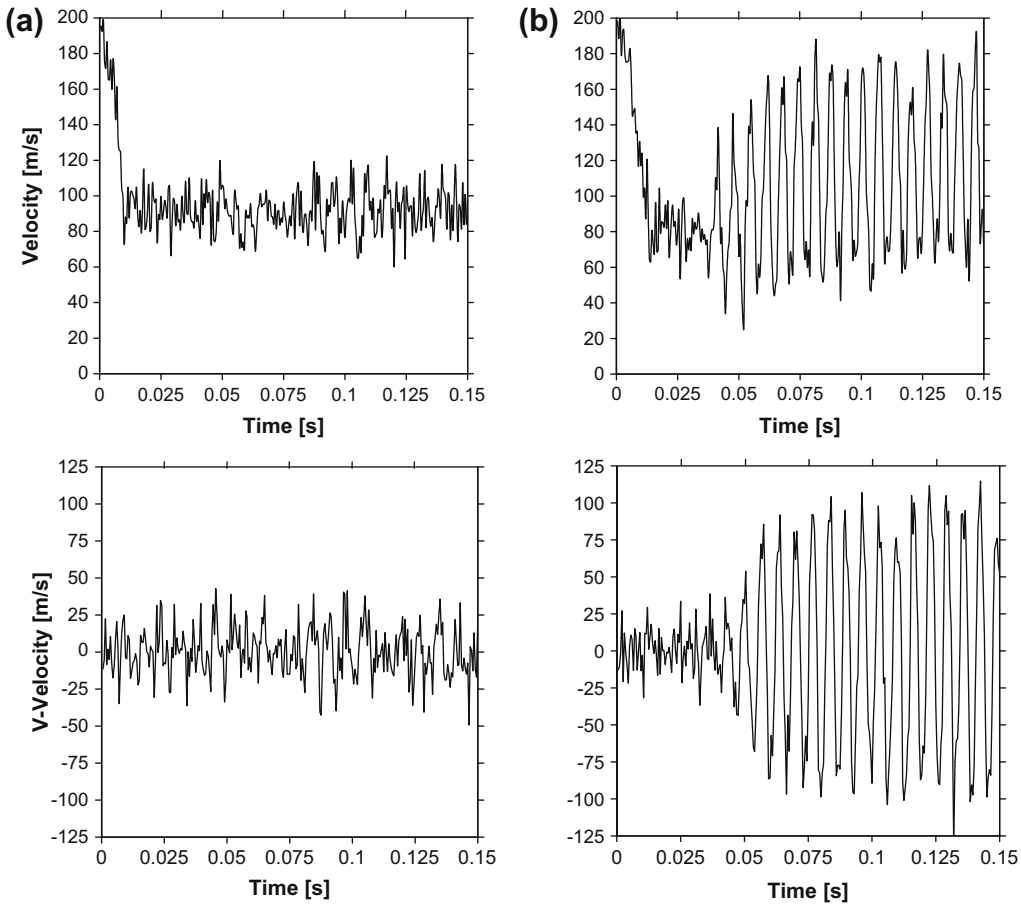

Fig. 15. Evolution of the *x*-(upper) and *y*-(lower) components of velocity (m/s) at sampling points S1 and S2 respectively (a) without transient adaptive sub-cells (TAS) and (b) with TAS.

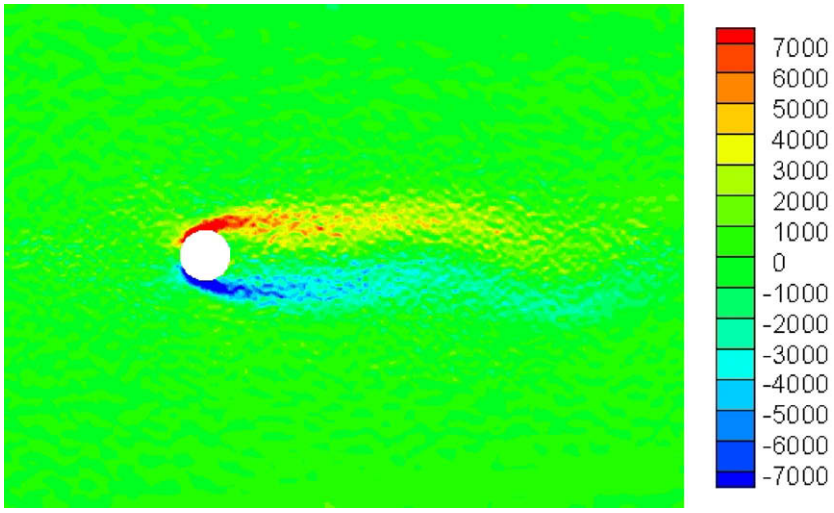

Fig. 16. Contours of vorticity (s^{-1}) at 0.1055 s for the vortex shedding behind a cylinder using transient adaptive sub-cells (TAS) and after post-processing by DREAM.

The Strouhal number and drag coefficient determined when TAS were employed were found to be 0.172 and 0.56, respectively. The values determined experimentally by Roshko [32] are 0.156 and 0.75, respectively. The reduction in drag coefficient is thought to be largely due to the experimental results being conducted in continuum conditions, whereas in the simulations the Knudsen number is in the range where significant velocity slip occurs around the cylinder. The simulation which did not employ TAS exhibits only slight periodic disturbances and it is difficult to determine a value for the Strouhal number.

Fig. 14 shows contours of the stream-wise component of velocity at several instants in the flow for the simulations with and without TAS. The data is also shown after post-processing using a modified DSMC Rapid Ensemble Averaging Method (DREAM) which was developed by Cave et al. [20]. In the modified version, the new particle data is generated using the original phase-space data rather than a Maxwell–Boltzmann velocity distribution. The plots clearly show that the magnitude of the fluctuations is considerably greater when sub-cells are employed. The reason for this is that the employment of sub-cells ensures that vorticity is preserved to the scale of the sub-cells rather than only the sampling cells. Post-processing using DREAM results in a reduction in the statistical scatter in the results.

Fig. 15 shows the normal and perpendicular components of velocity sampled behind the cylinder for the simulations with and without TAS. When sub-cells are employed these fluctuations for *y*-component velocity increase to a magnitude of approximately ±100 m/s whereas when sub-cells are not employed these fluctuation are greatly reduced (±30 m/s), are more irregular, do not build in magnitude, and can be attributed to random noise in the simulations.

Fig. 16 shows contours of vorticity around the cylinder for several time instances. Here the data has been post-processed using DREAM. The maximum vorticity occurs on the surface on the cylinder, due to large velocity difference between free-stream and the surface slip velocity, and is approximately 35,000 s^{-1}, although it is almost invisible in the plot since the region where it is located is very small.

5. Conclusions

A transient adaptive sub-cell technique appropriate for unstructured grids was proposed and implemented in the parallel-DSMC code (PDSC). Verification simulations of steady driven cavity flow, steady hypersonic flow over a two-dimensional cylinder, hypersonic flow over a cylinder/flare, and vortex shedding behind a cylinder show that the method is capable of reproducing benchmark results with greatly reduced computational expense, increases the accuracy of simulations by increasing the collision quality and preserves vorticity at the scale of the sub-cells. These procedures greatly enhance the capability of the PDSC code, enabling simulations to be conducted with under-resolved sampling cells. The method allows the range of parallel-DSMC simulations to be extended to unsteady, near-continuum flows and provides the basis of a virtual mesh refinement (VMR) technique which is currently under development.

References

[1] Bird GA. Molecular gas dynamics and the direct simulation of gas flows. Oxford: Clarendon Press; 1994.
[2] Bhatnagar PL, Gross EP, Krook M. A model collision process in gases. I. Small amplitude processes in charge and neutral one-component system. Phys Rev 1954;94:511–25.
[3] Yang JY, Huang JC. Rarefied flow computations using nonlinear model Boltzmann equations. J Comput Phys 1995;120(2):323–39.
[4] Bird GA. Monte-Carlo simulation in an engineering context. Prog Astro Aero 1981;74:239–55.
[5] Koura K, Matsumoto H. Variable soft-sphere molecular model for inverse-power or Lennard-Jones potential. Phys Fluids A 1991;3:2459–65.
[6] Nanbu K. Direct simulation scheme derived from the Boltzmann equation. I. Monocomponent gases. J Phys Soc Jpn 1980;49:2042–9.
[7] Wagner W. A convergence proof for Bird's direct simulation Monte Carlo method for the Boltzmann equation. J Stat Phys 1992;66:1011–44.
[8] Gallis MA, Torczynski JR, Rader DJ, Bird GA. Accuracy and convergence of a new DSMC algorithm. AIAA paper 2008-3913; 2008.
[9] Dietrich S, Boyd I. Scalar and parallel optimized implementation of the direct simulation Monte Carlo method. J Comput Phys 1996;126:328.
[10] Ivanov M, Markelov G, Taylor S, Watts J. Parallel DSMC strategies for 3D computations. In: Proc parallel CFD'96, Capri (Italy); 1997. p. 485.
[11] LeBeau GJ. A parallel implementation of the direct simulation Monte Carlo method. Comput Methods Appl Mech Eng 1999;174:319.
[12] Wu JS, Lian YY. Parallel three-dimensional direct simulation Monte Carlo method and its applications. Comput Fluids 2003;32(8):1133–60.
[13] Wu JS, Tseng KC, Wu FY. Parallel three-dimensional DSMC method using mesh refinement and variable time-step scheme. Comput Phys Commun 2004;162:166–87.
[14] Wu JS, Tseng KC. Parallel DSMC method using dynamic domain decomposition. Int J Numer Methods Eng 2005;63:37–76.
[15] Wu JS, Chou SY, Lee UM, Shao YL, Lian YY. Parallel DSMC simulation of a single under-expanded free orifice jet from transition to near-continuum regime. J Fluids Eng 2005;127:1161.
[16] Cave HM, Krumdieck SP, Jermy MC. Development of a model for high precursor conversion efficiency pulsed-pressure chemical vapor deposition (PP-CVD) processing. Chem Eng J 2008;135:120–8.
[17] Cave HM. Development of modelling techniques for pulsed pressure chemical vapour deposition (PP-CVD). Ph.D. Thesis, University of Canterbury; 2008.

[18] Wu JS, Lee F, Wong SC. Pressure boundary treatment in micromechanical devices using the direct simulation Monte Carlo method. JSME Int J Ser B 2001;44(3):439–50.
[19] Wu JS, Hsiao WJ, Lian YY, Tseng KC. Assessment of conservative weighting scheme in simulating chemical vapour deposition with trace species. Int J Numer Methods Fluids 2003;43:93.
[20] Cave HM, Tseng KC, Wu JS, Jermy MC, Huang JC, Krumdieck SP. Implementation of unsteady sampling procedures for the parallel direct simulation Monte Carlo method. J Comput Phys 2008;227(12):6249–71.
[21] Tseng KC, Wu JS, Boyd I. Simulations of re-entry vehicles by using DSMC with chemical-reaction module, AIAA-2006-8084. In: Proc 14th AIAA/AHI space planes and hypersonic systems and technologies conf, Canberra (Australia); 2006.
[22] LeBeau GJ, Jacikas KA, Lumpkin FE. Virtual sub-cells for the direct simulation Monte Carlo method. In: Proc of 41st AIAA aerospace sciences meeting and exhibit, Reno (NV); 2003.
[23] Bird GA. Sophisticated DSMC. Lecture notes from the short course at the DSMC07, Santa Fe (United States); September 30–October 3, 2007.
[24] Bird GA. Sophisticated versus simple DSMC. In: Proc. of 25th international symposium on rarefied gas dynamics, St. Petersburg (Russia); 2006.
[25] Lofthouse AJ, Boyd ID, Wright MJ. Effects of continuum breakdown on hypersonic aerothermodynamics. AIAA paper 2006-993; 2006.
[26] Holden MS. Measurement in regions of laminar shock wave/boundary layer interaction in hypersonic flow-code validation, CUBRC report (CD-ROM); 2003.
[27] Wang WL. A hybrid particle/continuum approach for nonequilibrium hypersonic flows. Ph.D. Thesis. University of Michigan; 2004.
[28] Meiburg E. Comparison of the molecular dynamics method and the direct simulation Monte Carlo technique for flows around simple geometries. Phys Fluids 1986;29(10):3107–13.
[29] Koura K. Direct simulation of vortex shedding in dilute gas flows past an inclined flat plate. Phys Fluids A: Fluid Dynam 1990;2(2):209–13.
[30] Bird GA. Knudsen and Mach number effects on the development of wake instabilities. AIAA paper 1998-785; 1998.
[31] Talbot-Stern J, Auld DJ. Direct simulation (Monte Carlo) of two dimensional vortex streets. In: Proc of AIAA/ASME joint thermophysics and heat transfer conference, Albuquerque (NM); 1998.
[32] Roshko A. On the Development of turbulent wakes from vortex streets, Ph.D. Thesis, California Institute of Technology; 1952.