ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

NUMERICAL ANALYSIS OF RAREFIED GAS FLOWS USING THE ACADEMIC CFD CODE GALATEA

Angelos G. Klothakis[1], Georgios N. Lygidakis[2] and Ioannis K. Nikolos[3]

[123]Technical University of Crete
School of Production Engineering and Management
University Campus, Chania, GR-73100, Greece
e-mail: [1]anklothakis@isc.tuc.gr,[2]glygidakis@isc.tuc.gr,[3]jnikolo@dpem.tuc.gr

Keywords: Rarefied Gas Flows, Navier-Stokes Equations, Velocity Slip Conditions, Temperature Jump Conditions, Direct Simulation Monte Carlo (DSMC).

Abstract. *During the last decades considerable efforts have been exerted for the development of micro air vehicles as well as microelectromechanical systems in general, for a wide range of applications. However, such systems involve microscale rarefied gas flows, which appear to be significantly different comparing to flows at macroscale and continuum regime; it is this the reason the Navier-Stokes equations fail to simulate such phenomena without further modification. To this end, the enhancement of the in-house academic Computational Fluid Dynamics solver Galatea to encounter such simulations is reported in this study. In case of rarefied gas flows and particularly for fluids in slip flow regime (Knudsen number greater than 0.01) the no-slip condition on solid wall surfaces is no longer valid; hence, velocity slip conditions as well as temperature jump ones have to be included instead. Furthermore, to increase accuracy at the same region the second-order accurate spatial slip model of Beskok and Karniadakis has been incorporated, which avoids the numerical difficulties, entailed by the evaluation of the second derivative of slip velocity when complex geometries along with unstructured hybrid grids are encountered. Due to oscillations that might appear, especially during the initial steps of the iterative procedure, a normalization scheme is additionally employed, to allow for the gradual increase of the corresponding slip/jump values. Galatea has been validated against a benchmark test case concerning rarefied laminar flow (inside the slip flow regime) over a wing with a NACA0012 airfoil in different angles of attack. The obtained results were compared with those of a reference solver, and with those obtained with the paralleld open-source kernel SPARTA, based on the Direct Simulation Monte-Carlo method. According to this last approach, the flow domain is divided into a finite number of computational cells, while the required sample macroscopic flow properties are retrieved assuming intermolecular collisions of the simulated particles inside such cells. An excellent agreement was achieved between the results obtained by Galatea and SPARTA as well.*

1 INTRODUCTION

During the past decade a significant effort has been exerted by various researchers for the development of micro air vehicles as well as microelectromechanical systems in general [1, 2, 3]. However, such systems may involve rarefied gas flows, which appear to be considerably different, compared to flows at the continuum regime; thus, the Navier-Stokes PDEs (Partial Differential Equations), used at macroscale CFD (Computational Fluid Dynamics) solvers, appear to fail simulating such phenomena without further adaptions and modifications [1]. In practice the rarefied gas flows are categorized depending on the computed Knudsen number, a classification originally proposed by Schaaf and Chambre [4]: For Knudsen numbers less than *1.0E-2* (continuum regime) the Navier-Stokes PDEs are valid without any further modification, allowing ordinary CFD (Computational Fluid Dynamics) solvers to be employed. Nevertheless, for values between *1.0E-2* and *1.0E-1* (slip flow regime) special treatment of wall boundary conditions is required; velocity slip conditions as well as temperature jump ones have to be applied (as in this work) [1]. If greater than *1.0E-1* Knudsen numbers are encountered (transition regime and free molecular regime) the rarefaction effects become the sovereign ones, necessitating for alternative methodologies, depending on the DSMC (Direct Simulation Monte Carlo) [5] approach or the solution of the Boltzmann Equation [6]. Figure 1 includes a schematic representation of the previously described classification. Although the latter method (based on the Boltzmann Equation) can actually be employed throughout all the regimes [1], the CFD approaches are usually preferred at continuum and slip flow ones; the implementation of Boltzmann Equation at those regimes calls for very large numbers of particles and consequently for excessive computational load and storage.

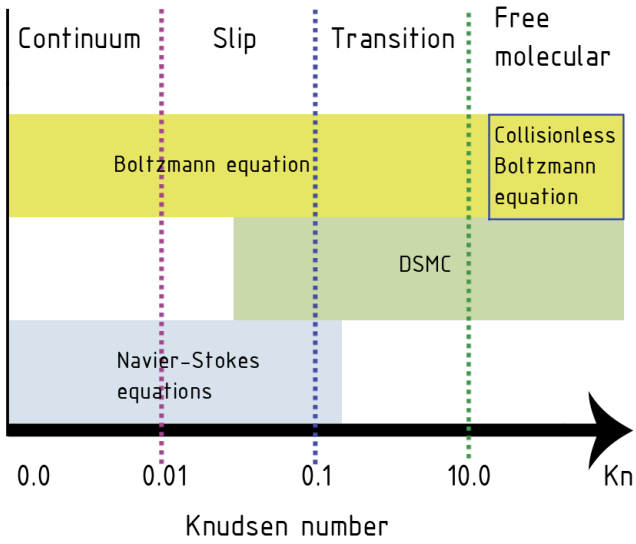


Figure 1: Classification of flow regimes

As mentioned above, in the slip flow regime (Knudsen number between *1.0E-2* and *1.0E-1*) the Navier-Stokes PDEs remain valid, although special treatment of solid wall boundaries is required. The no-slip condition cannot longer be applied on such surfaces; velocity slip conditions as well as temperature jump ones have to be imposed instead [1, 2, 3]. Despite the main concept of the aforementioned conditions was initially proposed by Navier [1], the first corresponding mathematical model was introduced by Maxwell in 1879 [7]; according to this approach the

values of slip velocity and temperature are defined with the corresponding normal to the boundary surface gradients [3]. Since then, various studies have been conducted on the prediction of the attitude of rarefied gases,focusing on the utilization of two- or three-dimensional computational fields, structured or unstructured grids, various geometries, etc., [8, 9, 10, 11, 12, 13]. Furthermore, significant efforts have been exerted for the development of higher-order spatial schemes to increase the accuracy of the final steady-state solution at the region of the solid wall boundaries [14, 15, 16, 17]. Among them the model of Beskok and Karniadakis appears to be especially attractive, as it avoids the numerical difficulties, entailed by the evaluation of the second derivative of slip velocity when complex geometries along with unstructured hybrid grids are encountered [14, 15]. As the aforementioned boundary conditions are imposed in a wall-function mode, they are susceptible to produce residual oscillations, especially during the initial steps of an explicit iterative solution; a remedy to this drawback was proposed by Ferras et al. [18], where a normalization scheme is considered, allowing for the gradual increase of the slip/jump values. Finally, few quite many studies include comparisons of the DSMC and the modified by the slip/jump boundary conditions Navier-Stokes PDEs approaches (as in this work), demonstrating the potential of such methodologies to effectively simulate rarefied gas flows at slip flow regime [1, 14, 19].

Despite the development of slip velocity and temperature jump boundary conditions allowed CFD solvers to extend their applications at slip flow regime, the latter appear to be inadequate to predict flows with larger values of Knudsen number (greater than *1.0E-1*) [1]. It is this observation along with the excessive computational requirements of methodologies depending on the Boltzmann equation that created the need for the development of the DSMC approach, originally introduced by Bird [5]. Mainly due to the statistical-stochastic nature of this modelling type, it wasn't initially accepted by the scientific community, confining its implementation in relatively simple simulations [5]. Nevertheless, since then it made its way, extending its applications in various engineering problems and demonstrating its capability for accurate flow predictions. The main drawback of the DSMC approach is the relatively excessive computational requirements in test cases involving high pressure and large-scale computational fields. This drawback was alleviated by the advance of the available computer systems and subsequently the capability of parallel processing. The first reported parallel DSMC approach (a NASA production code nowadays) was the DSMC Analysis Code (DAC), developed by LeBeau [20, 21], while since then quite many such algorithms have been developed, such as the SMILE [22], the MONACO [23], the ICARUS [24], the MGDS [25], the dsmcFoam [26] and the most recent one the open-source kernel SPARTA [27] (used also in this work).

In this study the enhancement of the in-house academic CFD solver *Galatea* to simulate rarefied gas flows inside the slip flow regime (Knudsen number between *1.0E-2* and *1.0E-1*) is reported [28]. *Galatea* employs the Navier-Stokes PDEs in dimensionless formulation along with a node-centered finite-volume scheme to predict inviscid, viscous laminar, and viscous turbulent flows (with RANS-Reynolds Averaged Navier-Stokes approach) of compressible fluids on tetrahedral or hybrid unstructured grids [28]. For rarefied gases the capability of implementing the aforementioned slip velocity and temperature jump boundary conditions [1] on solid wall surfaces has been incorporated. In order to increase the accuracy at the same regions, a second-order accurate slip scheme has been included. Particularly, the slip model of Beskok and Karniadakis [14, 15] was adopted due to its relatively easy implementation on unstructured, tetrahedral or hybrid grids; it can overcome the numerical difficulties, entailed by the evaluation of the second derivative of slip velocity [14]. Furthermore, the normalization scheme, reported in [18], was included to mitigate the excessive oscillations, caused by the Dirichlet-type of the

slip/jump boundary conditions, especially during the initial steps of the iterative solution procedure. The proposed solver was validated against a benchmark test case concerning rarefied laminar flow over a wing with a NACA0012 airfoil in different angles of attack [13, 19, 29]. The obtained results have been compared with those of another reference solver, confirming its potential for such simulations. In addition, further evaluation of the *Galatea* algorithm was succeeded with the parallel open-source kernel SPARTA [27], a DSMC code as mentioned above. An excellent agreement was achieved between the results obtained by *Galatea* and SPARTA as well.

The structure of this paper is as follows: In Section 2 the *Galatea* solver is described in brief, while the main attention is directed towards the analysis of the incorporated slip velocity and temperature jump conditions. Section 3 contains an outline of the DSMC approach, as well as of the employed open-source kernel SPARTA. In Section 4 the extracted by *Galatea* and SPARTA results against a test case considering rarefied laminar flow over a rectangular wing in different angles of attack are presented and compared, while Section 5 includes the corresponding conclusions and some information on ongoing work.

2 THE CFD APPROACH

2.1 The *Galatea* code

In this study the in-house academic CFD code, named *Galatea* [28], was used, enhanced though with appropriate modifications to account for rarefied gas flows. The proposed solver employs a node-centered finite-volume method on three-dimensional unstructured grids, composed of tetrahedral, prismatic and pyramidical elements, to simulate inviscid, viscous laminar and viscous turbulent compressible fluid flows. For turbulence prediction the RANS approach is implemented along with appropriate statistical two-equation models, namely the $k - \varepsilon$ [30], the $k - \omega$ [31] and the SST (Shear Stress Transport) [32] models. An upwind method is implemented for the computation of the inviscid fluxes, applying the Roe's [33] or the HLLC (Harten-Lax-van Leer-Contact) [34] approximate Riemann solvers, coupled with a higher-order accurate spatial scheme, based on the MUSCL (Monotone Upwind Scheme for Conservation Laws) approach along with appropriate slope limiters, namely the Van Albada-Van-Leer [35], the Min-mod [36], the Barth-Jespersen [37] and the MLP-Venkatakrishnan (Multi-dimensional Limiting Process-Venkatakrishnan) [38, 39] limiter. For the computation of the viscous fluxes the required velocity and temperature gradients are evaluated with an element-based (edge-dual volume) or a nodal-averaging method [28], either selected by the user. Time integration and iterative approximation of the final steady-state solution is succeeded with either an explicit scheme, applying a second-order temporal accurate four-stage Runge-Kutta (RK(4)) method, or an implicit one, implementing the Jacobi or the Gauss-Seidel algorithm [28]. In order to increase the efficiency of the proposed solver, appropriate acceleration techniques have been incorporated, such as an edge-based data structure [40], a local time-stepping technique [40], parallel processing based on the domain decomposition approach and the MPI (Message Passing Interface) library functions [28], and an agglomeration multigrid scheme [28, 41]. Finally, an h-refinement technique has been included to increase accuracy at pre-selected regions of the examined grid, by enriching them with more nodes during the solution procedure, while simultaneously avoiding the generation of dense meshes from the very beginning [28]. Further details for the *Galatea* solver can be found in [28] and [41].

2.2 Velocity slip and temperature jump conditions

As mentioned in Introduction, the no-slip conditions on solid wall surfaces appear to fail in the slip flow regime; velocity slip and temperature jump wall functions have to substitute them [1, 19]. Although their main idea was initially captured by Navier [1], the first such model was reported by Maxwell in 1879 [7]. According to this approach, the values of slip velocity and temperature on wall boundaries are defined with the corresponding normal to the boundary surface gradients. A schematic representation of the previous state for slip velocity values is illustrated in Figure 2; moreover it highlights the difference between the no-slip and the slip boundary conditions. Mathematically, the Maxwell model is described for the dimensionless slip velocity U_s as follows [1, 7, 12, 14]

$$U_s - U_w = \frac{2-\sigma_u}{\sigma_u} Kn \frac{\partial U_s}{\partial n} + \frac{3}{2\pi} \frac{(\gamma-1)}{\gamma} \frac{Kn^2 Re}{E_c} \frac{\partial T}{\partial s} \tag{1}$$

where U_w denotes the velocity on the wall surface, K_n is the Knudsen number and $\frac{\partial U_s}{\partial n}$ the transverse velocity gradient, i.e., the derivative of the tangential slip velocity normal to the wall surface (denoted by vector n) [12]. Further, σ_u is the tangential momentum accommodation coefficient (equal to unity in this study) [13, 19], which actually models the momentum exchange of the gas molecules impinging on the solid boundaries. It depends on the surface quality; small values of this constant can considerably increase the slip velocity even for low Knudsen numbers [14]. The second term includes the values of the dimensionless Reynolds Re and Eckert E_c numbers, along with the ideal gas constant γ and the tangential to the wall surface derivative of the temperature. It actually represents the slip velocity contribution induced by the thermal creep; it appears usually to be relatively small, due to the second-order in Knudsen number as well as due to the fact that in common engineering problems there is a small temperature change across the wall surfaces; therefore, in this study it is neglected [12].

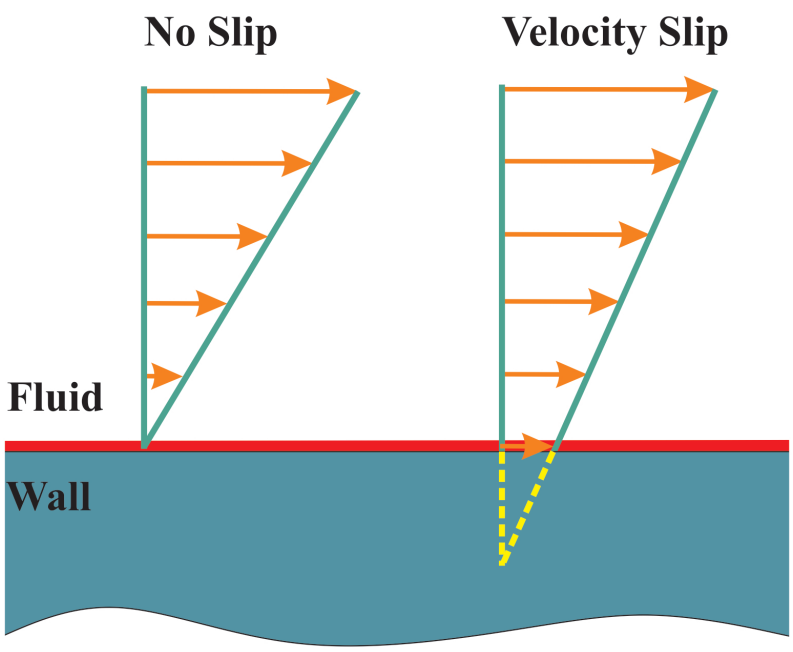


Figure 2: Velocity slip boundary conditions.

The temperature jump boundary conditions are defined similarly to the slip velocity ones as [12]

$$T_s - T_w = \frac{2-\sigma_T}{\sigma_T}\frac{2\gamma}{\gamma+1}\frac{Kn}{Pr}\frac{\partial T}{\partial n} \tag{2}$$

where T_s represents the slip temperature, T_w the wall temperature and $\frac{\partial T}{\partial n}$ the derivative of the temperature normal to the wall surface. Pr is the dimensionless Prandtl number, while σ_T is the thermal or energy accommodation coefficient; similarly to the momentum coefficient the thermal one depends on the surface quality [14], while in this work it was set equal to unity [13, 19].

In order to increase the accuracy of the final steady-state solution at the solid walls' region, several different second-order accurate spatial schemes have been reported in the open literature [14, 15, 16, 17]. Their main concept is based on the reconstruction of the first term of Equation (1) with the Taylor series [16, 17]. Nevertheless, such a reconstruction is not a straightforward procedure when complex three-dimensional geometries, along with unstructured hybrid grids, are encountered; the calculation of the second derivative of the tangential velocity in the normal direction to the surface may introduce numerical errors, despite the *a priori* computational difficulty it entails in such test cases [14]. Therefore, in this study the approach, introduced by Beskok and Karniadakis [14] was adopted, according to which the second-order accurate slip velocity can be computed as

$$U_s - U_w = \frac{2-\sigma_u}{\sigma_u}\frac{Kn}{(1-bKn)}\frac{\partial U_s}{\partial n} \tag{3}$$

where b is the slip coefficient, which can be defined either experimentally or from other methodologies employing the Boltzmann equation or the DSMC approach [14]. It actually represents the vorticity flux into the surface divided by the vorticity of the flow field on the surface, obtained by the no-slip approximation; in this study it was set equal to *-1.0* [14].

The velocity slip and temperature jump boundary conditions are implemented in a Dirichlet way, hence they are susceptible to produce oscillations during the iterative procedure (especially during the initial steps of an explicit iterative scheme), or even lead simulation to fail [18]. Therefore, a normalization scheme was incorporated in *Galatea* solver, allowing for the gradual increase of the slip velocity and temperature values; mathematically, it is described as follows [18]

$$U_s^i = \alpha U_s^{i-1} + (1-\alpha)U_s^i \tag{4}$$

where α is the normalization coefficient, taking values smaller than unity; in this work it was set equal to *0.95*. In case a multigrid accelerated simulation is performed, the aforementioned slip velocity and temperature values are computed only at the finest resolution, while at next they are restricted to the coarser multigrid levels similarly to the rest nodal values [41].

3 THE DSMC APPROACH

3.1 The particle method

The DSMC method, originally introduced by Bird [5], is a stochastic-type scheme using a large number of particles to simulate gas flows, especially rarefied ones. Each of the aforementioned particles, called simulator particle, actually represent actually a large number of real particles [5]. According to the main idea of this methodology the computational domain is divided in Cartesian cells, which provide the geometric boundaries and volumes required to

sample the macroscopic properties of the flow. The latter are acquired taking into account the movements and collisions of the simulator particles; therefore, the time step has to be defined in a way allowing the movement and collisions of each particle to be decoupled. In conclusion, the DSMC algorithm can be divided in four main steps: a) Given a time step the particles move to their new positions. b) They are indexed into the computational cells. c) The collision pairs are selected and the corresponding intermolecular collisions are performed on a probabilistic manner. d) Since the temporary steady-state solution has been achieved, the required macroscopic flow properties, such as velocity and temperature, are obtained by averaging temporally (for several time steps) the microscopic properties of the particles in the cells [5]. The pre-described procedure is presented in the flow chart of Figure 3.

3.2 The SPARTA DSMC kernel

As mentioned in Introduction, the parallel open-source kernel SPARTA is used in this study, where SPARTA stands for Stochastic Parallel Rarefied-gas Time-Accurate Analyzer. It was developed in Sandia Laboratories by Gallis and Plimpton [27] and distributed as an open source code under the terms of the GPL license. It is capable of both serial and parallel processing, the latter based on the domain decomposition approach and the MPI communication protocol (similarly to the *Galatea* solver). Despite its robustness and flexibility it is not addressed to non-experienced users, as it calls for a thorough understanding of the DSMC approach; it executes by reading the commands in the script line by line [27].

3.2.1 Problem definition

In order to run a simulation with SPARTA the corresponding parameters have to be defined first, i.e., the simulation box (computational domain), the grid, the internal boundaries (inside the simulation box), the particle species (e.g., oxygen, nitrogen, ions) and the initial population of the particles. For a two-dimensional run, the simulation box is defined by the coordinates of the boundaries; minimum and maximum coordinates in x- and y- axis. For a three-dimensional simulation the coordinates of two more boundaries have to be defined (including coordinates in z-axis). As far as the computational discretization is concerned, SPARTA employs a hierarchical Cartesian grid strategy. The simulation domain is considered initially as a aingle grid cell (level 0). The aforementioned cell is divided in N_x by N_y by N_z cells at level 1, while at next each of these cells can be further divided into smaller ones at different levels. The properties of the particle species, such as their diameter and molecular weight, are acquired from an additional appropriately formatted file. Finally, the initial population of the particles is defined from the *Fnum* number, denoting the number of real particles represented by a simulator particle. As far as initialization of the procedure is concerned, the corresponding velocities are retrieved from a Maxwellian speed distribution function.

3.2.2 Settings

Since the problem definition stage is accomplished the settings of the input script have to be defined. In this section various parameters are determined, i.e., the collision model, the load balancing (static at the beginning and dynamic during the iterative procedure), the boundary conditions, the surface reflection models, the sample averaging-type and the desired simulation outputs [27]. This stage is of extreme importance, as it influences the accuracy of the final results as well as the required computation time. For example, the time step, the used flow

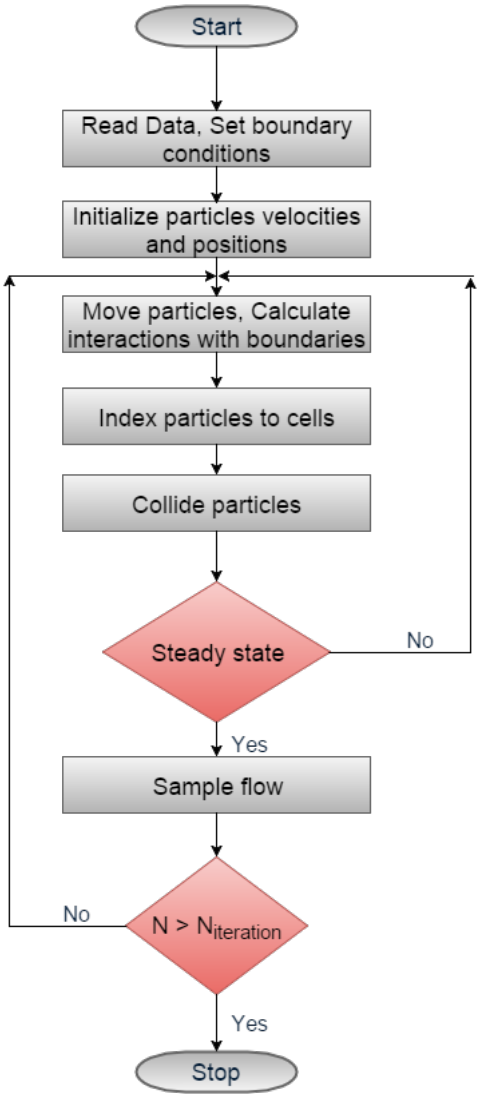


Figure 3: The DSMC Method

field variables, the characteristics of the boundary surfaces, and the sampling intervals included in this script have a serious impact on the extracted solution. Especially the latter parameter, defining the intervals at which the macroscopic flow properties will be extracted, appears to be very important, as an inappropriate value can lead to incorrect results or to an efficiency drop.

Furthermore, at this stage dynamic load-balancing type is defined. SPARTA can adjust the assignments of grid cells and their corresponding particles to each processor in order to spread more efficiently the computational load. Since the dynamic load-balancing type is determined, the cells assigned to each processor are either clumped or dispersed. Actually, there are three ways the aforementioned balancing can be succeeded; it can be performed depending on clumped assignments of child cells to each processor, dispersed assignments of child cells to each processor, or on a recursive coordinate bi-sectioning approach [27]. According to the first method each grid cell is assigned randomly to one of the processors; as a result, the processors do not get necessarily the same amount of grid cells. If the second strategy is followed, each processor selects randomly another processor to assign its first grid cell to; at next, it loops over its grid cells and assigns each of them to consecutive processors. Unfortunately, this method-

Case	$Mach_\infty$	Re_∞	Kn_∞	V_∞, m/s	α, deg	ρ_∞, kg/m^3	T_∞, K	T_w, K
A	2.0	106	0.026	506	0	6.026×10^-5	161	290
B	2.0	106	0.026	506	10	6.026×10^-5	161	290

Table 1: Flow conditions for Cases A and B.

Number density, m^-3	F_{num}	Time-step, sec	Transient period	Sample period
1.296×10^{21}	1.2×10^{15}	3.0×10^{-7}	140000	40000

Table 2: Parameters used for the SPARTA code runs.

ology suffers from the same to the first approach shortcoming. Finally, according to the third scheme, the recursive coordinate bisectioning method (RCB), the processors are assigned compact clumps of grid cells [27]. For every cell a weight coefficient is defined, hence, dynamic load balancing aims to assign equal total weights to all the available processors. This weight coefficient can be defined in various ways, e.g., if it is set equal to unity, each processor is assigned the same number of cells, while if it is set equal to the number of the included particles, the same number of particles is assigned to all the processors. Dynamic load-balancing is strongly influenced by two more parameters, namely the imbalance factor and the frequency interval. The first one defines the maximum number of particles run by a processor divided by the average number of particles per processor, while the second the frequency in which the imbalance of the simulation is checked for change. Both the aforementioned coefficients have to be wisely selected in order communication overheads to be avoided during the simulation. For the test cases encountered in this study, an imbalance factor of *20%-30%* (defined as *1.2* or *1.3* in the input script) was found to be sufficient enough, without causing additional communication load, while the imbalance was checked for possible change two times during the simulation.

4 NUMERICAL RESULTS

The *Galatea* solver was validated against a benchmark test case considering rarefied laminar compressible fluid flow over a rectangular wing with a NACA0012 airfoil [13, 19, 29]. The Mach number was assumed equal to *2.0*, while the Knudsen and Reynolds numbers equal to *0.026* and *106*, respectively [19]; the latter was based on the mean aerodynamic chord, which was redimensionalized to become equal to unity due to the dimensionless formulation of the proposed code. Two different angles of attack were examined, namely *0*° (Case A) and *10*° (Case B) [13, 19, 29]. Freestream temperature was assumed equal to $161K$, while the corresponding value on the wing surface equal to $290K$. The aforementioned flow conditions are summarized in Table 1; for *Galatea* solver simulations they were used though in dimensionless formulation [28]. For the representation of the computational field a hybrid unstructured grid was constructed, composed of *305,978* nodes, *566,245* tetrahedra and *394,760* prisms, the latter located on the solid wall region (wing) to allow for the effective prediction of the boundary layer region. Figure 4 illustrates the employed computational grid, while Figure 5 depicts its density on the symmetry surface.

For the computation of the inviscid fluxes the Roe's approximate Riemann solver along with a second-order spatial accurate scheme, coupled with the Min-mod limiter, was employed; the corresponding viscous fluxes were computed based on the nodal-averaging scheme. Time inte-

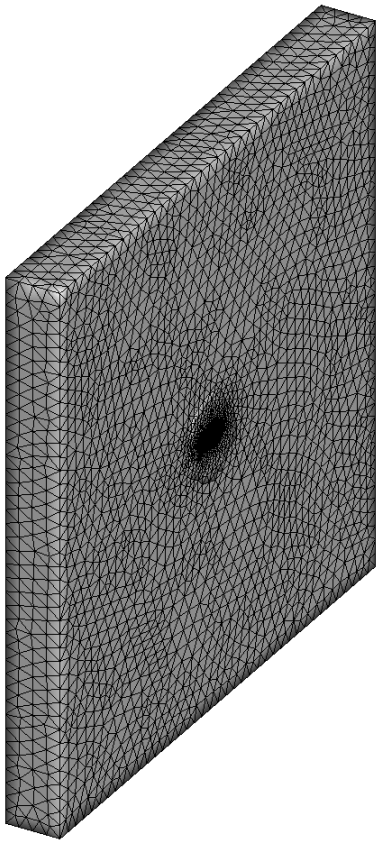


Figure 4: Computational grid used with the *Galatea* code.

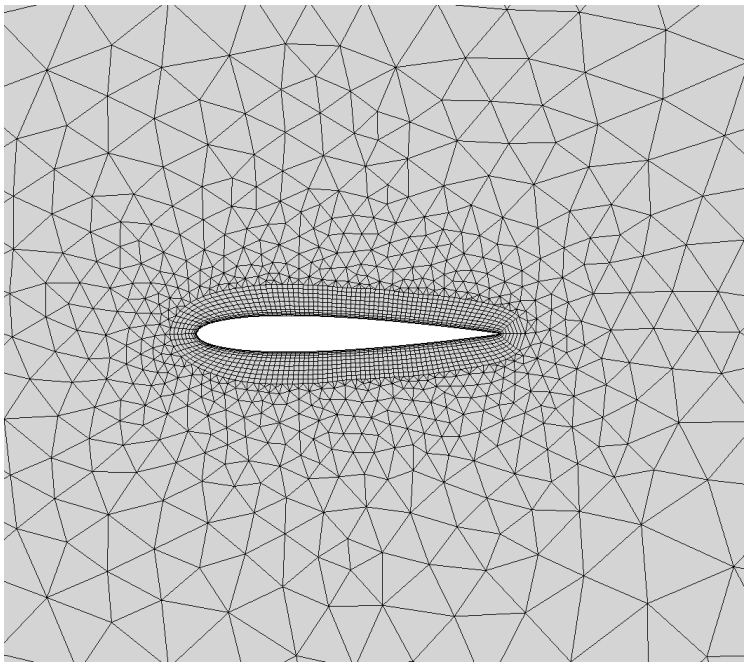


Figure 5: Density on the symmetry surface of the computational grid used with the *Galatea* code.

gration was succeeded via the incorporated explicit scheme (Runge-Kutta method) with a CFL number equal to *0.5*. Both the incorporated first- and second-order spatial accurate slip models were implemented [14]. However, no difference was identified between their results; this lack of difference is attributed to the relatively fine grid utilized. To accelerate the solution procedure on a (DELL^{TM} R815 Poweredge Server with four AMD Opteron^{TM} 6380 sixteen-core processors) the initial grid was decomposed in four sub-domains to be processed in parallel; two coarser levels were constructed for each of them applying the full-coarsening directional agglomeration strategy, in order the incorporated multigrid scheme to be applied [41].

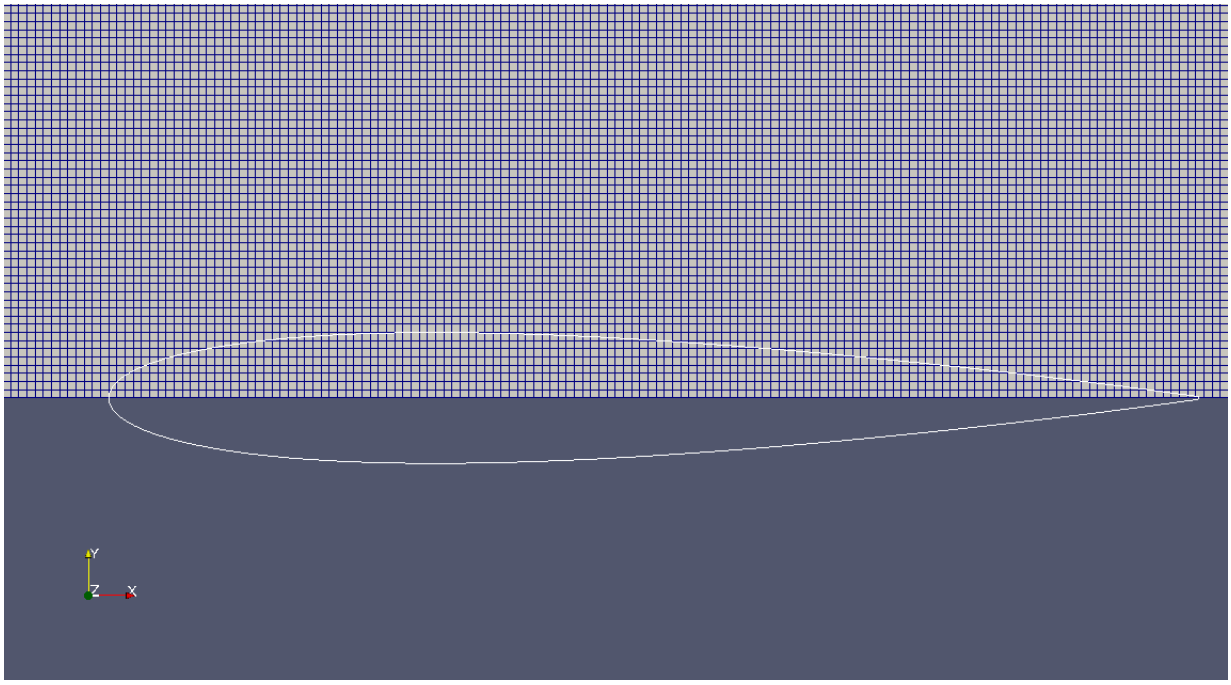
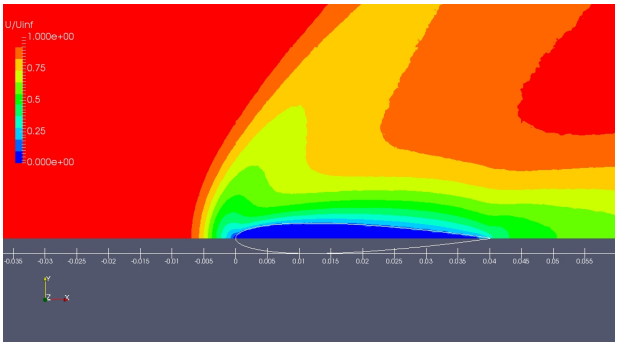


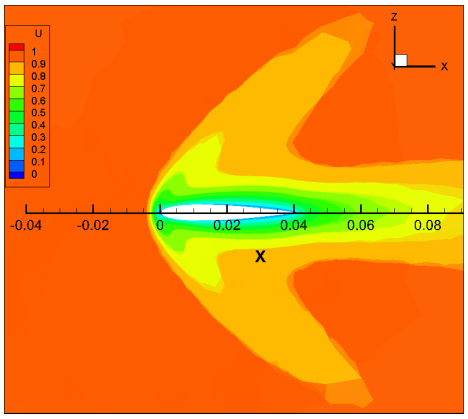
Figure 6: Computational grid used with the parallel open-source kernel SPARTA.

As mentioned in Introduction, the evaluation of the proposed solver was succeeded with a DSMC approach code, the parallel open-source kernel SPARTA [27]. The encountered computational domain was designed with dimensions *0.38m* and *0.24m* along the x- and y-axis, respectively; its discretization was succeeded with a Cartesian two-dimensional mesh, composed of *1200* cells along x-axis and *800* cells along y-axis. Considering that the mean free path in such test cases is approximately *0.000923m*, each grid cell was constructed to be more or less *1/3* the size of it, a common practice in such simulations [5]. Furthermore, the grid around the airfoil was generated such as no more than four points, used for airfoil representation (a total number of *1600* was used), to be included at each computational cell; as a result, a smoother discretization was achieved. Figure 6 illustrates a close-up view of the utilized computational grid around the NACA0012 airfoil.

The SPARTA code was run on the same to *Galatea* solver computer system with the flow conditions summarized in Table 1. As far as the simulation parameters are concerned, the number density was set equal to $1.296 \times 10^{21}\ m^{-3}$, while the time step equal to $3.0 \times 10^{-7} s$; the latter value along with the employed grid density ensures that each particle needs approximately three time steps to cross entirely a grid cell [5]. The simulation began with a transient period of *140,000* steps, producing the initial steady-state solution, while at next samples were taken for

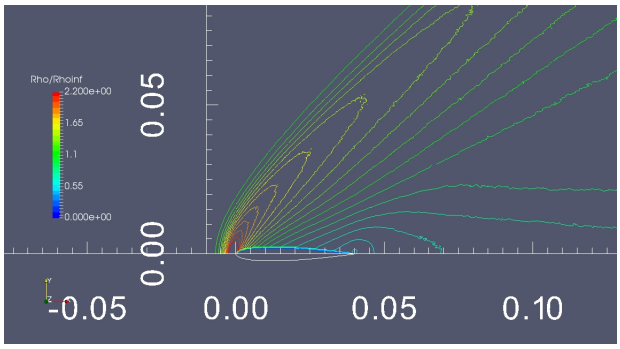


(a) SPARTA.

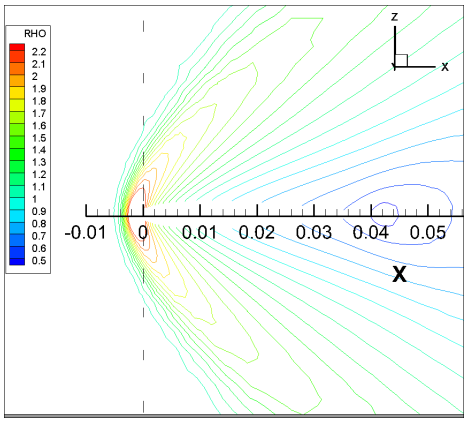


(b) *Galatea*.

Figure 7: Velocity contours (Case A).



(a) SPARTA.



(b) *Galatea*.

Figure 8: Density contours (Case A).

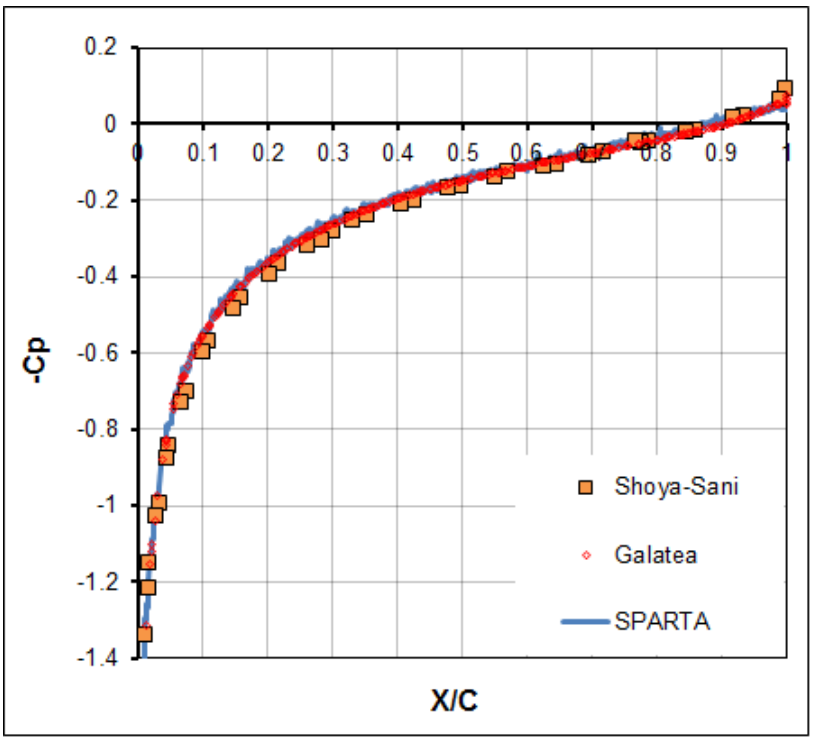


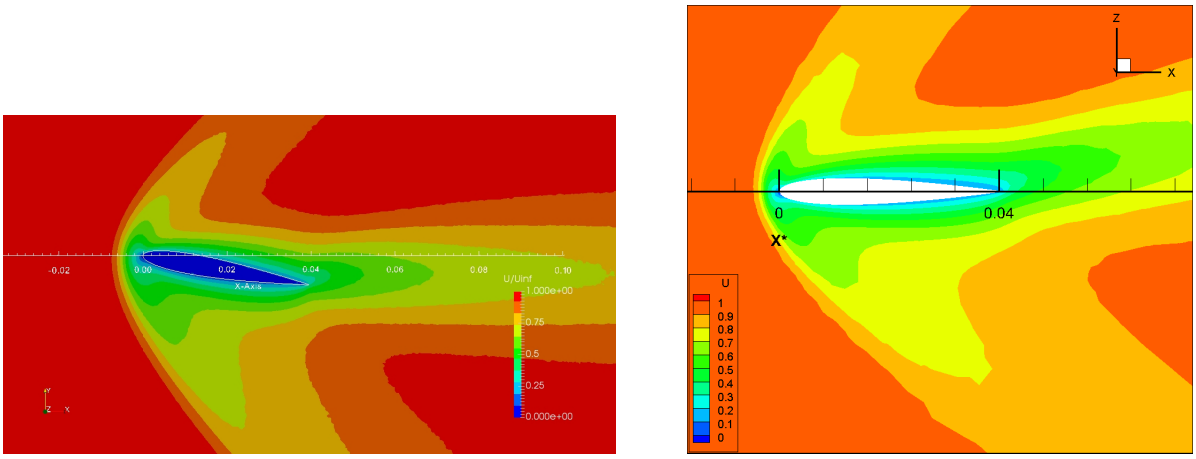
Figure 9: Distribution of pressure coefficient *Cp* around the NACA0012 airfoil (Case A).

additional *40,000* steps, aiming to reduce the statistical scattering error. The aforementioned parameters are outlined in Table 2. Considering that DSMC simulations depend strongly on the employed number of particles, several different numbers of them were tested (between 2×10^6 and 16×10^6) along with the aforementioned grid, as well as with a finer one; the latter was generated by refining each computation cell around the airfoil into *5x5* smaller ones. From this study the optimum *Fnum* (real particles per one simulator particle) of 1.2×10^{15} was derived, which subsequently produced a total number of 8×10^6 simulator particles for the whole simulation domain. As far as the grid is concerned, no difference was identified, hence, the initial mesh was selected. The previous parameters defining actually the DoFs (Degrees of Freedom) of the simulation were proved to produce the desired accuracy, avoiding though any excessive requirements for computational load and storage.

In Figure 7 the extracted velocity contours around the NACA0012 airfoil by both the *Galatea* and SPARTA solvers are presented for the first test case (Case A - angle of attack *0°*). As one can observe a satisfactory comparison is obtained, especially in the bow shock and stagnation regions, indicating qualitatively the potential of the proposed solver for such simulations. Furthermore, a very good agreement can be observed with the available experimental and numerical results, reported in [29] and [13, 19] respectively. The same similarity can be identified in Figure 8, depicting the corresponding density contours. It should be highlighted that dimensionless values are included in both the aforementioned figures. Figure 9 includes the derived distributions, compared with the one reported in [19]; the distribution of *Galatea* code simulation was extracted from the mid-span of the wing geometry.

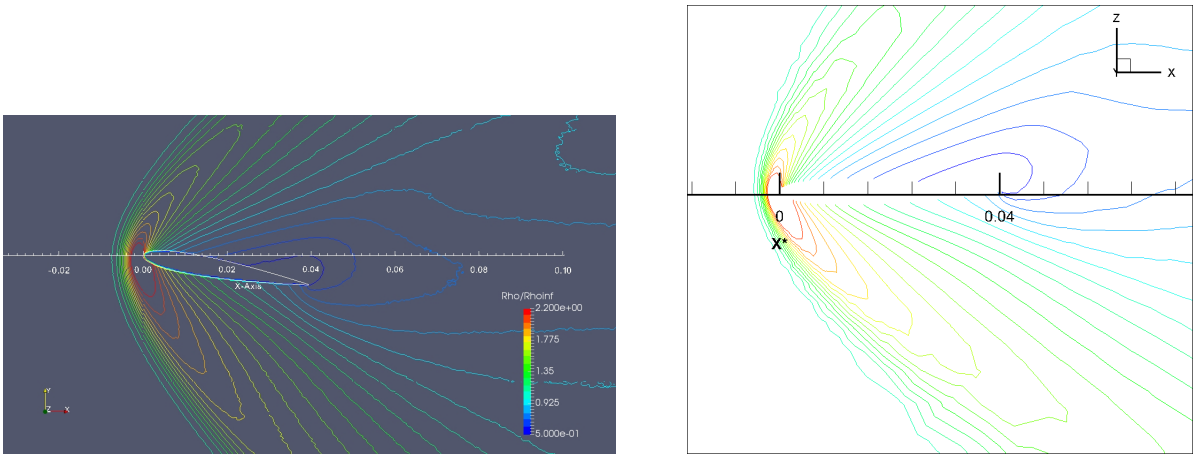
Similarly to Case A, Figure 10 illustrates the extracted velocity contours around the NACA0012 airfoil by both the *Galatea* and SPARTA codes for Case B (angle of attack *10°*). Once more a very good qualitative agreement is succeeded between the results of the employed solvers,

while the same holds in Figure 11 presenting the corresponding density contours. A quantitative comparison for the distributions of the pressure coefficient *Cp* around the NACA0012 airfoil, depicted in Figure 12. It is obvious that for both test cases (with angle of attack *0°* and *10°*) the employed codes produce almost identical results, despite they are based on completely different numerical approaches.



(a) SPARTA. (b) *Galatea*.

Figure 10: Velocity contours (Case B).



(a) SPARTA. (b) *Galatea*.

Figure 11: Density contours (Case B).

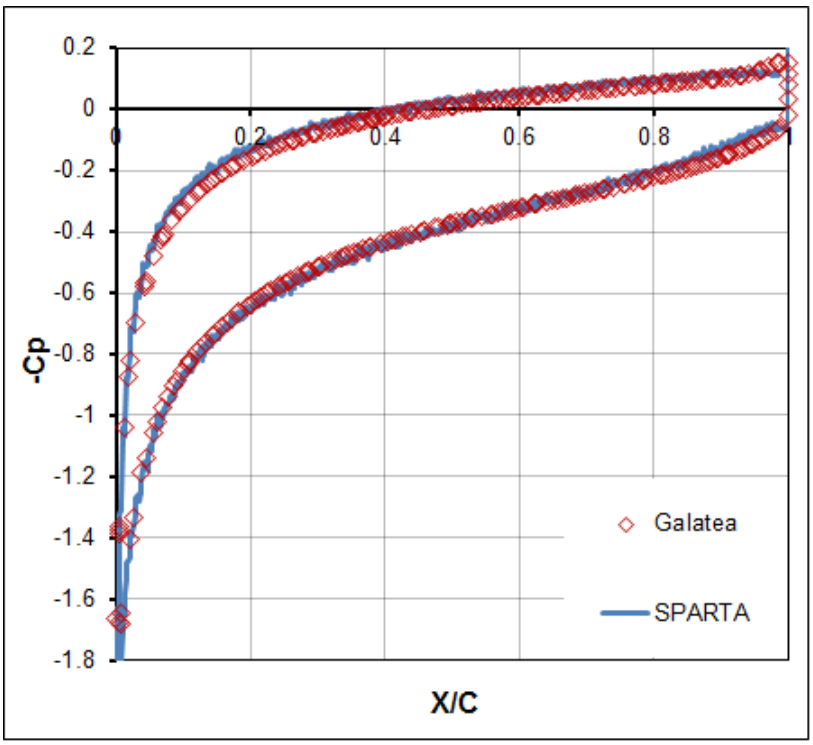


Figure 12: Distribution of pressure coefficient *Cp* around the NACA0012 airfoil (Case B).

5 CONCLUSIONS

In this study the enhancement of the in-house academic CFD solver *Galatea* to simulate rarefied gas flows in slip flow regime (Knudsen number between *1.0E-2* and *1.0E-1*) is reported [28]. Velocity slip and temperature jump models have been incorporated to this end, as the no-slip boundary conditions imposed on solid wall surfaces in flows in continuum regime appear to fail when rarefied gases are examined. The incorporated second-order accurate model didnt produce improved results; this lack of difference is attributed to the relatively fine grid utilized. In addition, the incorporated normalization scheme [18] was revealed a considerable enhancement, as the Dirichlet-type of the slip/jump boundary conditions was found to cause excessive oscillations without it, especially during the initial steps of the iterative solution procedure. The proposed solver was validated against a benchmark test case concerning rarefied laminar flow over a wing with a NACA0012 airfoil in different angles of attack [13, 19, 29]. The parallel open-source kernel SPARTA [27] was used for comparison purposes, while a very good agreement between the results of the two codes was achieved. Finally, ongoing work includes further assessment of the enhanced *Galatea* solver against more complex test cases of rarefied gas dynamics, as well as further examination of the incorporated second-order accurate slip model, utilizing coarser grids.

6 ACKNOWLEDGEMENTS

This work has been co-funded by the European Commission (European Regional Development Fund) and the Greek State, through the Operational Programme ”Competitiveness and Entrepreneurship (OPCE II 2007-2013), National Strategic Reference Framework - Research funded project: Development of a spark discharge aerosol nanoparticle generator for gas flow

visualisation and for fabrication of nanostructured materials for gas sensing applications DE_ SPARK_ NANO_ GEN", 11SYN_5_144, in the framework of the Action "COOPERATION 2011" — Partnerships of Production and Research Institutions in Focused Research and Technology Sectors.

REFERENCES

[1] W.M. Zhang, G. Meng and X. Wei, A review on slip models for gas microflows, *Microfluidics and Nanofluidics,* 2012.

[2] C.M. Ho and Y.C. Tai, Micro-electro-mechanical-systems (MEMS) and fluid flows, *Annual Review of Fluid Mechanics,* vol. 30, pp. 579-612, 1998.

[3] M. Gad-el-Hak, The fluid mechanics of microdevices The Freeman scholar lecture, *ASME Journal of Fluids Engineering,* vol. 121, 1999.

[4] S.A. Schaaf and P.L. Chambre, Flows of rarefied gases, *Princeton University Press,* Princeton, 1961.

[5] G.A. Bird, Molecular gas dynamics and the direct simulation of gas flows, *Clarendon Press,* Oxford, 1994.

[6] C.R. Lilley and J.E. Sader, Velocity profile in the Knudsen layer according to the Boltzmann equation, *Proceedings of the Royal Society A,* vol. 464, pp. 2015-2035, 2008.

[7] J.C. Maxwell, On stresses in rarified gases arising from inequalities of temperature, *Philosophical Transactions of the Royal Society,* vol.170, pp.231-256, 1879.

[8] N. Dongari, R. Sambasivam and F. Durst, Extended Navier-Stokes equations and treatments of micro-channel gas flows, *Journal of Fluid Science and Technology,* vol. 4, pp. 454-467, 2009.

[9] R.S. Myong, A computational method for Eu's generalized hydrodynamic equations of rarefied and microscale gasdynamics, *Journal of Computational Physics,* vol. 168, pp. 47-72, 2001.

[10] R.S. Myong, A generalized hydrodynamic computational model for rarefied and microscale diatomic gas flows, *Journal of Computational Physics,* vol. 195, pp. 655-676, 2004.

[11] R.S. Myong, J.M. Reese, R.W. Barber and D.R. Emerson, Velocity slip in microscale cylindrical Couette flow: The Langmuir model, *Physics of Fluids,* vol. 17, 2005.

[12] H.D. Madhawa Hettiarachchi, Mihajlo Golubovic, William M. Worek and W.J. Minkowycz, Three-dimensional laminar slip-flow and heat transfer in a rectangular microchannel with constant wall temperature, *International Journal of Heat and Mass Transfer,* vol. 51, pp. 5088-5096, 2008.

[13] J. Fan, I.D. Boyd, C.P. Cai, K. Hennighausen and G.V. Candler, Computation of rarefied gas flows araound a NACA 0012 airfoil, *AIAA Journal,* vol. 39, pp. 618-625, 2001.

[14] A. Beskok and G.E. Karniadakis, A model for flows in channels, pipes, and ducts at micro and nano scales, *Microscale Thermophysical Engineering,* vol. 3, pp. 43-77, 1999.

[15] G.E. Karniadakis and A. Beskok, Micro flows: fundamentals and simulation, *Springer,*New York, 2002.

[16] C. Shen, Rarefied Gas Dynamics. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[17] N.D. Stevanovic and V.D. Djordjevic, An exact analytical solution for the second order slip-corrected Reynolds lubrication equation, *FME Transactions,* vol. 43, pp. 16-20, 2015.

[18] W.L. Ferras, J.M. Nobregal and F.T. Pinho, Implementation of slip boundary conditions in the finite volume method: new techniques, *International Journal for Numerical Methods in Fluids,* vol. 72, pp. 724-747, 2013.

[19] A. Shoja-Seni, E. Roohi, M. Kahrom and S. Stefanov, Investigation of aerodynamic characteristics of rarefied flow around NACA0012 airfoil using DSMC and NS solvers, *European Journal of Mechanics B/Fluids,* vol. 48, pp. 59-74, 2014.

[20] G.J. LeBeau, A parallel implementation of the direct simulation Monte Carlo method, *Computer Methods and Applications in Mechanical Engineering,* vol. 174, no. 3, pp. 319337, 1999.

[21] M.A. Gallis, K.A. Boyles, G.J. LeBeau, DSMC simulations in support of the STS-107 accident investigation, *Proceedings of the 24th Rarefied Gas Dynamics (RGD) Symposium,* Bari, Italy, July 2004.

[22] M.I.G. Markelovf and S.F. Gimelshein, Statistical simulation of reactive rarefied flows: numerical approach and applications, *Proceedings of the 7th AIAA/ASME Joint Thermophysics and Heat Transfer Conference,* Albuquerque, June 1998.

[23] S. Dietrich and I.D. Boyd, Scalar and parallel optimized implementation of the direct simulation Monte Carlo method, *Journal of Computers and Physics,* vol. 126, no. 2, pp. 328342, 1996.

[24] T.J. Bartel, S. Plimpton, and M.A. Gallis, Icarus: A 2-D Direct Simulation Monte Carlo (DSMC) code for multi-processor computers, *Sandia National Laboratories Technical Report,* 2001-2901, Albuquerque.

[25] D. Gao and T.E. Schwartzentruber, Optimizations and OpenMP implementation for the Direct Simulation Monte Carlo method, *Journal of Computers and Fluids,* vol. 42, no. 1, pp. 7381, 2011.

[26] T.J. Scanlon, E. Roohi, C. White, M. Darbandi, and J.M. Reese, An open source, parallel DSMC code for rarefied gas flows in arbitrary geometries, *Journal of Computers and Fluids,* vol. 39, no. 10, pp. 20782089, 2010.

[27] M.A. Gallis, J.R. Torczynski, S.J. Plimpton, D.J. Rader, and T. Koehler, Direct Simulation Monte Carlo: The quest for speed, *Proceedings of the 29th Rarefied Gas Dynamics (RGD) Symposium,* Xi'an, China, July 2014.

[28] G.N. Lygidakis and I.K. Nikolos, Numerical analysis of flow over the NASA Common Research Model using the academic Computational Fluid Dynamics code Galatea, *ASME Journal of Fluids Engineering,* vol. 137, 2015.

[29] J. Allegre, M. Raffin and J.C. Lengrand, Experimental flow fields around NACA0012 airfoils located in subsonic and supersonic rarefied air streams, *M.O. Bristeau et al. (eds.), Numerical Simulation of Compressible Navier-Stokes Flows,* pp. 5968, 1985.

[30] B.E. Launder and D.B. Spalding, The numerical computation of turbulent flows, *Computer Methods in Applied Mechanics and Engineering,* vol. 3, pp. 269289, 1974.

[31] P.G. Saffman and D.C. Wilcox, Turbulence-model predictions for turbulent boundary layers, *AIAA Journal,* vol. 12, pp. 541546, 1974.

[32] F.R. Menter, Two-equation eddy viscosity turbulence models for engineering applications, *AIAA Journal,* vol. 32, pp. 1598-1605, 1994.

[33] P. Roe, Approximate Riemann solvers, parameter vectors and difference schemes, *Journal of Computational Physics,* vol. 43, pp. 357-371, 1981.

[34] E.F. Toro, M. Spruce and W. Speares, Restoration of the contact surface in the HLL Riemann solver, *Shock Waves,* vol. 4, pp. 25-34, 1994.

[35] G.D. Van Albada, B. Van Leer and W.W. Roberts, A comparative study of computational methods in cosmic gas dynamics, *Astronomy and Astrophysics,* vol. 108, pp. 46-84, 1982.

[36] P.K. Sweby, High resolution schemes using flux limiters for hyperbolic conservation laws, *SIAM Journal on Numerical Analysis,* vol. 21, pp. 995-1011, 1984.

[37] T.J. Barth and D.C. Jespersen, The design and application of upwind schemes on unstructured meshes, *27th AIAA Aerospace Sciences Meeting and Exhibit,* Reno, USA, January 9-12, 1989, AIAA-89-0366.

[38] V. Venkatakrishan, Convergence to steady state of the Euler equations on unstructured grids with limiters, *Journal of Computational Physics,* vol. 118, pp. 120130, 1995.

[39] J.S. Park, S.H. Yoon and C. Kim, Multi-dimensional limiting process for hyperbolic conservation laws on unstructured grids, *Journal of Computational Physics,* vol. 229, pp. 788-812, 2010.

[40] J. Blazek, Computational Fluid Dynamics: Principles and Applications. *Elsevier Science,* Kidlington, 2001.

[41] G.N. Lygidakis, S.S. Sarakinos and I.K. Nikolos, Comparison of different agglomeration multigrid schemes for compressible and incompressible flow simulations, *Advances in Engineering Software,* in press, 2016.