Numerical Simulation of High-Speed Air Flows with High-Temperature Effects on Graphics Processor Units

Vladislav N. Emelyanov[1], Anton G. Karpenko[2], Konstantin N. Volkov[3]

[1] Baltic State Technical University, St Petersburg, Russia, vlademelyanov@gmail.com
[2] Saint Petersburg State University, St Petersburg, Russia, aspera.2003.ru@mail.ru
[3] Kingston University, London, United Kingdom, k.volkov@kingston.ac.uk

Abstract

Numerical simulation of shock wave phenomena in supersonic compressible flows with high-temperature effects is discussed. The finite volume method is applied to solve unsteady three-dimensional compressible Navier–Stokes equations on unstructured meshes. High-temperature gas effects altering the gas dynamical processes are taken into account. Possibilities of the use of graphics processor units (GPUs) for the simulation of high-speed flows are demonstrated. Solutions of some test cases on GPUs are reported, and a comparison between computational results of equilibrium chemically reacting and perfect air flowfields is performed. Speedup of solution on GPUs with respect to the solution on central processor units (CPUs) is compared. The results obtained provide promising perspective for designing a GPU-based software framework for practical applications.

1 Introduction

Shock waves appear in nature and technological processes whenever the different elements in a fluid approach one another with a velocity higher than the local speed of sound. Disturbances, which propagate wavelike in a continuum, represent singularities of the set of functions of the flow and the variables of state of the medium and can be distinguished both from the point of view of their mathematical description and physical meaning. The moving discontinuity changes continuously, while sweeping across the fluid mass. The formation of a shock wave is dependent on the objects that affect the flowfield. Dissipation of energy, rapid changes in velocity, pressure, temperature and flow turning are some of the features associated with shock waves.

The aero-thermodynamic design of hypersonic vehicles incorporates wind tunnel testing, flight experiments and computer simulation [1]. Many aerodynamic and propulsion characteristics still remain uncertain and are difficult to predict due to the lack of flight test data and the limitations of ground test facilities [2]. The ground-based experimental facilities are not able to reproduce full-scale complex physical and chemical phenomena conditions for flight due to short duration of useful test times [3]. The methods of computational fluid dynamics (CFD) are particularly attractive due to its relatively low cost and its ability to deliver data that cannot be measured or observed [4, 5]. The quality of CFD calculations of the flows strongly depends on the proper prediction of flow physics (non-equilibrium effects, radiation, molecular dissociation, chemical reactions).

High-temperature gas effects are a key issue related to hypersonic aerodynamic design and are currently poorly to moderately modeled in CFD [6]. Thin shock layers due to high compression, entropy layers caused by highly swept and curved shock waves, viscous/inviscid interactions and real gas effects are all complex flow features that occur in hypersonic flow. There is also restriction in the speed of computers and a limitation of reliable quantitative data about high-temperature constants.

Speed and accuracy are key factors in the evaluation of CFD solver performance. Advanced simulation tools enable increased spatial and time accuracy, geometric complexity, mesh adaptation, physical process complexity, uncertainty quantification and error control. Unstructured meshes present more flexibility and higher accuracy to represent problems that have complex geometries and boundaries. For enhanced spatial accuracy, high-order algorithms for unstructured meshes are developed. High performance computing (HPC) resources are widely used in

In: Sergey I. Smagin, Alexander A. Zatsarinnyy (eds.): V International Conference Information Technologies and High-Performance Computing (ITHPC-2019), Khabarovsk, Russia, 16-19 Sep, 2019, published at http://ceur-ws.org

engineering applications. The stagnation in the clock-speed of central processing units (CPU) has led to significant interest in parallel architectures that offer increasing computational power by using many separate processing units. Modern graphics processing units (GPU) provide architectures and new programming models that enable to harness their large processing power and to design CFD simulations at high performance and low cost [7, 8]. Achieving good performance for unstructured mesh based CFD solvers on GPUs is more difficult due to their data dependent and irregular memory access patterns [9].

To simulate supersonic flows and reproduce real physical processes, it is necessary to use high-order numerical methods minimizing the numerical viscosity to the level smaller than the physical viscosity, or methods that do not produce numerical dissipation. In the case where a low accuracy scheme is used, its pattern is smoothed due to the numerical effects and does not represent the real structure of the flow.

Although a large number of non-equilibrium simulation codes have been developed, the implementation of the numerical solvers is not considered as fully completed because of limited information with regard to physical model (chemical reactions, transport models) and also validation (lack of both numerical solutions and experimental results for comparison). These methods require extensive computer processor time and storage, and are not generally applicable to parametric and optimization studies or preliminary design calculations. The complex physical phenomena and a wide range of spatial and time scales present in hypersonic flows making the development of efficient and accurate numerical simulation methods challenging.

The motivation of this paper is to assess the in-house compressible CFD code for hypersonic flow simulations and to demonstrate successful design of a highly parallel computation system based on GPUs. The unstructured CFD code uses an edge-based data structure to give the flexibility to run on meshes composed of a variety of cell types. Possibilities of the use of GPUs for the simulation of high-speed and high-temperature flows are discussed. The results obtained are generally in reasonable agreement with the available experimental and computational data.

2 Mathematical Model

Mathematical model used for numerical simulation of hypersonic flows has to capture the complex thermo-physical phenomena that characterize these flows. Governing equations are formulated for a one-temperature model with eleven species. The unsteady three-dimensional flow of the viscous compressible gas is described with a system of equations including mass conservation equation, momentum conservation equation and energy conservation equation. The Sutherland's law is used to obtain molecular viscosity as a function of temperature. Sutherland's law is accurate for air over a range of several thousand degrees and is appropriate for hypersonic viscous flow calculations. The thermal conductivity is obtained from dynamic viscosity and the Prandtl number.

Vibrational and chemical processes take place as a result of molecular collisions. The actual number of collisions required depends on the type of molecule and the relative kinetic energy between the two colliding particles. In turn, molecular collisions take time to occur. The precise amount of time depends on the molecular collision frequency Z~p/T1/2, where p is the pressure and T is the temperature. Hence, the collision frequency is low for low pressures and very high temperatures. The equilibrium systems assumed that the gas has enough time for the necessary collisions to occur and that the properties of the system at a fixed pressure and temperature are independent on time (the reactions take place at equal rates in their direct and reverse directions).

The chemical equilibrium assumption consists of considering that all the species in the system have time enough to reach the equilibrium state. In calculations, air is represented as a mixture of 11 different ideal gases (e^-, N, O, Ar, N2, O2, NO, N+, O+, Ar+, N++, O++, Ar++).

In chemically equilibrium reacting flows, the chemical components computation is time consuming. The explicit expressions derived in [10] for equilibrium air are based on density and internal energy as the independent variables. These expressions are used to determine the other thermodynamic and transport properties, avoiding calculating chemical species. The model proposed in [10] takes into account the dissociation of oxygen and nitrogen, formation of nitrogen oxide, possibility of the appearance of an electronic component due to the single and double ionization of oxygen, nitrogen and argon. The oxygen dissociation reaction takes place, followed by reactions of nitrogen dissociation and single ionization of a weighted average mixture of these gases. When considering each subsequent reaction, the previous one is considered fully completed. In the temperature range from 200 to 20000 K and pressures from 0.001 to 1000 atm, the error of the model does not exceed 1.5% by density and 3% by enthalpy.

3 Computational Algorithm

The non-linear solver works in an explicit time-marching fashion, based on a Runge–Kutta stepping procedure. The unstructured CFD code uses an edge-based data structure to give the flexibility to run on meshes composed of a variety of cell types. The fluxes through the surface of a cell are calculated on the basis of flow variables at nodes at either end of an edge, and an area associated with that edge (edge weight). The edge weights are pre-computed and take into account geometry of the cell. The flux vector is split into the inviscid and viscous components. The governing equations are solved with upwind finite difference scheme for inviscid fluxes, and central difference scheme of the second order for viscous fluxes. For simulation of low-speed flows, convergence to a steady state is accelerated by the use of low-Mach number preconditioning method. The computational procedure involves reconstruction of the solution in each control volume and extrapolation of the unknowns to find the flow variables on the faces of control volume, solution of Riemann problem for each face of the control volume, and evolution of the time step.

The computational procedure is implemented as a computer code in C/C++ programming language. Parallelization of the computational procedure is performed by a message passing interface (MPI). CUDA technology is used to implement GPU version of the code. An equivalent solver is made in C++ to be run in a CPU for benchmarking purposes.

4 Parallel Algorithm

A GPU architecture implements different types of memory for storing data (global memory, constant memory, texture memory, shared memory and registers). This memory structure allows to reduce global memory accesses and collaboration among threads in the same thread block. In terms of latency, global memory access is the slowest whereas registers are the fastest. Since the GPU execution model requires that the information is first placed in global memory and then accessed by the GPU application, it is necessary to optimize global memory access. Global memory access is optimized by achieving peak bandwidth and by reducing the number of accesses.

Explicit time-marching algorithms are the most convenient ones to be ported on to the GPU. This is because there is no iteration, and the new value of a variable depends only on the old time values. Hence, the update of a given variable is done independent of variables being updated on other threads. There is no recursive relation between the variables on the threads, since they are all known at the old time step. However, even for explicit algorithms, a few changes are needed for efficiently implementation of numerical algorithms on the GPU. These relate to the use of shared memory and the layout of data structures. Memory coalescing and block size influence the speed achieved. The data should be organized such that adjacent threads access adjacent nodal data. In addition, data should be, where possible, copied to shared memory and re-used as much as possible. Therefore, even explicit algorithm based CFD codes need to be reorganized to take advantage of the GPU architecture [9].

The performance critical portion of the CFD solver consists of a loop which repeatedly computes the time derivatives of the conserved variables. The conserved variables are then updated using an explicit Runge–Kutta time-stepping procedure. The most expensive computation consists of accumulating _ux contributions across each face when computing the time derivatives. Therefore, the performance of the CUDA kernel which implements this computation is crucial in determining whether or not high performance is achieved.

The implementation is split between CPU and GPU. Pre- and post-processing steps are done on the CPU, leaving only the computation itself to be performed on the GPU. For example, the CPU constructs the mesh and evaluates the face areas, face normals and cell volumes. The initialization of the flowfield is also done on the CPU. Each time step of the computation then involves a series of kernels on the GPU which evaluate the cell face fluxes, sum the fluxes into the cell, calculate the change in properties at each node, smooth the variables and apply boundary conditions.

To compare the performance of the CFD solver on CPU and GPU, similar algorithms for calculating the perfect gas and equilibrium air are implemented on CPU. Calculations are made on two different servers. Server 1 uses one core of the Xeon E5-2680 v3 CPU or one NVidia Tesla K40 calculation module. Server 2 uses one IBM Power8 CPU thread or one NVidia Tesla K40 module. Some characteristics of the servers used in CFD calculations are presented in the Table 1.

Table 1: Parameters of GPU harware.

Server	Server 1		Server 2	
Brand name	Huawei RH2288H		IBM S822LC	
RAM per node, GB	128		256	
Number of CPUs	2		2	
Number of GPUs	2		2	
Processor unit	Xeon E5-2680 v3	Tesla K40	IBM Power8	Tesla P100
Production year	2014	2013	2016	2016
Frequency (base), MHz	2500	745	2860	1328
Number of cores	12	2880	10	3584
Peak performance, GFlops (single precision, base)	960	4291	458	9519
Peak performance, GFlops (double precision, base)	480	1430	229	4760
Memory bus width, bit	64	384	64	4096

5 Result and Discussion

The GPU version of the CFD code is used and validated for a variety of benchmark test cases. These problems provide insight into the ability of a scheme to capture the bow shock, smoothly resolve the post-shock stagnation region flow and predict a smoothly varying heating distribution around the stagnation point.

5.1 Composition of Air

Composition of high-temperature air is shown in the Figure 1 and 2 where the equilibrium composition of high-temperature air (in terms of mole fraction) is given as a function of temperature at 0.01 atm. The oxygen begins to dissociate above 2100 K and is completely dissociated above 7800 K. The nitrogen begins to dissociate above 6000 K and is completely dissociated above 12000 K. The nitrogen oxide is present between 800 and 11000 K, with a peak mole fraction occurring about 2500 K. The line for atomic oxygen has a local maximum around 4000 K and then decreases at higher temperatures in the range from 6000 to 20000 K. The data plotted are in a good agreement with the detailed tabulations found in [11].

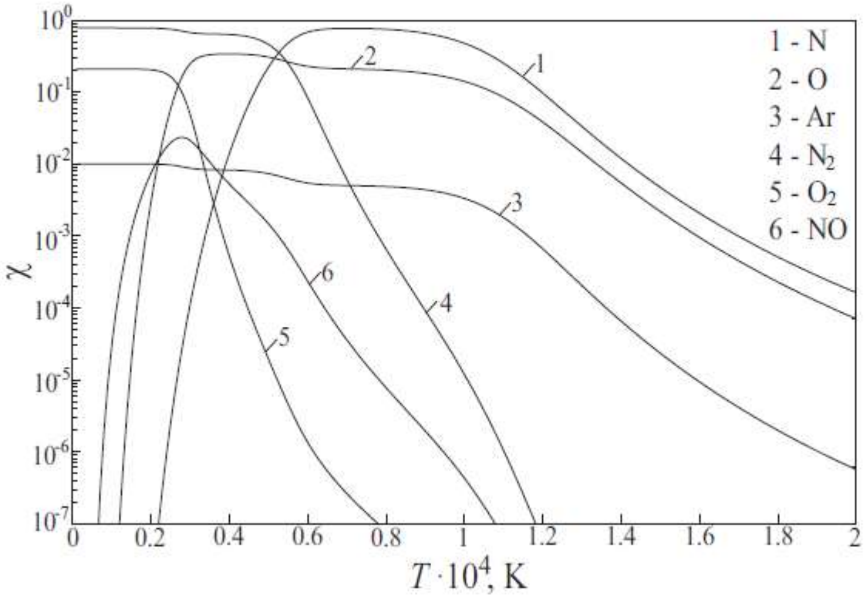

Figure 1: Composition of equilibrium air as a function of temperature at pressure 0.01 atm (1 — N, 2 — O, 3 — Ar, 4 — N_2, 5 — O_2, 6 — NO)

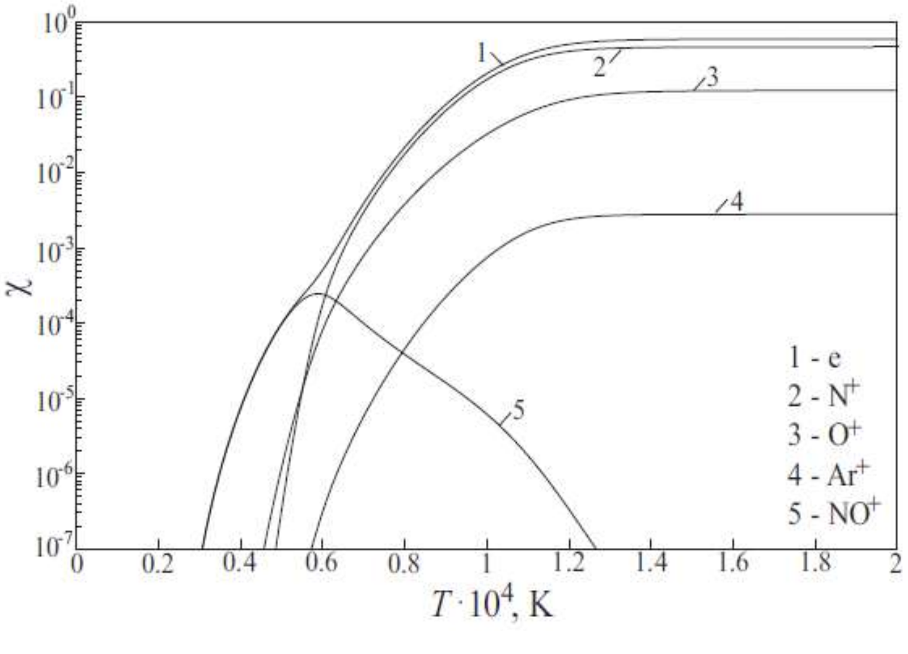

Figure 2: Composition of equilibrium air as a function of temperature at pressure 0.01 atm (1 — e^-, 2 — N^+, 3 — O^+, 4 — Ar^+, 5 — NO^+)

If the pressure is increased to 10 atm then all of the lines would qualitatively shift to the right, that is the various dissociation processes would be delayed to higher temperatures. On the other hand, if pressure is decreased, then all of the lines would qualitatively shift to the left, that is dissociation would occur at lower temperature. Hence, raising the pressure decreases the amount of dissociation, and lowering the pressure increases the amount of dissociation.

5.2 Shock Tube Problem

The shock tube test case considers a long tube (L=10 m) containing a gas separated by a thin diagram located at x=4 m. The gas is assumed to be at rest on both sides of the diagram (uL=uR=0), but it has different constant pressures (pL$=10^5$ Pa and pR$=10^5$ Pa) and densities (ρL=1.0 kg/m3 and ρR=0.125 kg/m3) on each side. The mixture of perfect gases chosen is 0.7778 N2 and 0.2222 O2 by mass fraction. This composition is closely match the analytical model for air at these temperatures. It will be a test of the perfect gas mixture if it can recreate the solution computed analytically. At time t=0, the diagram is removed, and the problem is to determine ensuing motion of the gas. The solution of this problem consists of a shock wave moving into the low pressure region, a rarefaction wave that expands into the high pressure region, and a contact discontinuity which represents the interface.

Time of calculation of 1000 time steps and speedup of calculations are presented in the Table 2 (time is given in seconds). Two indices are used to specify computational option. The first index corresponds to the time-marching scheme used in calculations based on one-step (option A) or three-step (option B) Runge–Kutta time-stepping procedure. The second index corresponds to the exact Godunov (index 1) or approximate Roe (index 2) Riemann solvers

Table 2: Time (in seconds per 1000 time steps) and speedup for shock tube problem.

No	Mesh 1			Mesh 2		
	CPU	GPU	SpeedUp	CPU	GPU	SpeedUp
A1	18.79	1.36	13.83	198.96	10.98	18.12
A2	27.79	1.60	17.35	261.63	13.31	19.65
B1	36.83	2.63	13.98	388.74	21.25	18.29
B2	56.28	3.14	17.90	515.97	26.28	19.63
No	Mesh 3			Mesh 4		
	CPU	GPU	SpeedUp	CPU	GPU	SpeedUp
A1	1918.32	99.60	19.26	17485.10	874.99	19.98
A2	2285.81	120.88	18.91	20542.40	914.83	22.46
B1	3776.11	197.58	19.11	35257.70	1736.47	20.30
B2	4672.10	240.95	19.39	40595.40	1822.97	22.27

Unstructured tetrahedral mesh is used to solve 3D shock tube problem. Calculations are based on different meshes. The coarsest mesh contains about 104 cells (mesh 1), and the finest mesh contains about 107 cells (mesh 4). The intermediate meshes contain 105 cells (mesh 2) and 106 (mesh 3) cells. The time step is 1.53×10-5 s, and the total computational time is 7.63×10-3 s. Courant number is equal to 0.85. The calculations are performed on one module of Tesla S1070 platform with 1.44 GHz (a number of cores is 256), and one core of CPU AMD Phenom 2 with 3 GHz.

The computed solution shows good agreement with the analytical solution. This is a fair indication that the mixture of perfect gases model has been implemented correctly. The discrepancies near the shock are the standard numerical errors involved with any finite-volume solution. Most of the discrepancies occur where the flow changes most rapidly. This is to be expected in any scheme where discontinuities are captured over a few cells that have large flow gradients.

5.3 Shock Wave Reflection from a Wall

Two types of reflection of shock waves are observed in a steady state supersonic flow: regular reflection (two-wave configuration) and Mach reflection (three-wave configuration). There are two possible solutions and the related hysteresis phenomenon. The location of the shock waves depends on free stream Mach number, angle of incidence of the shock wave, and ratio of specific heat capacities. The most studied case is the case of reflection of an oblique shock wave from a plane wall or a reflection of a plane shock wave from a wedge.

The problem of the reflection of an oblique shock wave incident at an angle φ to the horizontal axis from the wall is equivalent to the problem of the interaction of a direct shock wave with a wedge of 90-φ angle. A number of researches consider the reflection of a shock wave from the plane of symmetry in the interaction of shock waves generated by two symmetrical wedges placed in a supersonic flow. In contrast to the reflection from a solid wall, in this formulation of the problem the effects associated with the presence of a boundary layer are excluded.

A regular reflection of an oblique shock wave from a wall is considered when the angle of incidence of the shock wave does not exceed a certain threshold value. On the left and upper boundaries, the parameters of uniform flows corresponding to the flow in front of and behind the front of the oblique shock wave are given. The wave comes to the wall and is reflected from it at some angle. The flow consists of incident and reflected shock waves. The parameters ahead of the front of the incident wave and behind its front approximately correspond to the problem formulated in [12]. The Euler equations are solved in the region [0,4]×[0,1] shown in Figure 3. A shock wave initiated by the appropriate boundary conditions falls at an angle of 60 degrees to the horizontal axis. The calculations are performed on a structured mesh containing 2×10^6 nodes.

Time of calculation of 1000 time steps and speedup of calculations are presented in the Table 3 (time is given in seconds). Time discretization is based on the third-order Runge–Kutta scheme. Flux on the faces of the control volume are calculated with Godunov and Rusanov schemes. A perfect gas model and Kraiko model are used as thermodynamic model of air.

Table 3: Time (in seconds per 1000 time steps) and speedup for shock wave reflection problem.

Air model	Solver	Server 1			Server 2		
		CPU	GPU	Speedup	CPU	GPU	Speedup
Perfect gas	Godunov	48	1.63	29.41	55.00	0.30	186.44
	Rusanov	29	1.25	23.26	37.00	0.28	133.09
Kraiko model	Godunov	622.00	3.48	178.58	1032.00	0.69	1497.82
	Rusanov	618.00	2.85	216.69	1011.20	0.67	1520.60

5.4 Flow around Sphere

The shock standoff distance on spheres in hypersonic flow is one of the most appropriate parameter for validating the CFD results. Since for high Mach number flows on spheres, the shock standoff distance is much smaller than the body radius, its experimental determination is difficult and large errors have to be accepted. Some significant theoretical studies have been made to predict the shock stand-off distance on spheres in hypersonic flow.

The flow past a sphere with a diameter 12.7 mm with a hypersonic air flow and equilibrium chemical reactions is considered. The flow parameters correspond to those used in experiments [13]. The inlet flow pressure is 666.61 Pa, and the inlet temperature is 293 K (the density is 7.9 × 10-3 kg/m). The Mach number varies in the range from 7.1 to 17.77, which corresponds to the flight speed from 2438.4 to 6705.6 m/s.

Figure 3 shows that the shock layer thickness reduces with increasing free stream Mach number (increasing density behind the shock). Line 1 shows the shock standoff distance for air in full chemical equilibrium. Line 3 gives the shock standoff distance for a perfect gas with $\gamma =$ 1.4. Since the calculated perfect gas normalized density values tend towards a finite maximum value of about 6 for air as the free stream Mach number increases, the shock wave thickness tends towards a minimum. For the Mach 10 free stream, the shock layer thickness is already close to this minimum since the normalized density is about 5.7. For the same free stream conditions, the chemical equilibrium gas shock layer is thinner than that for perfect gas and is continually changing with increasing free stream Mach number.

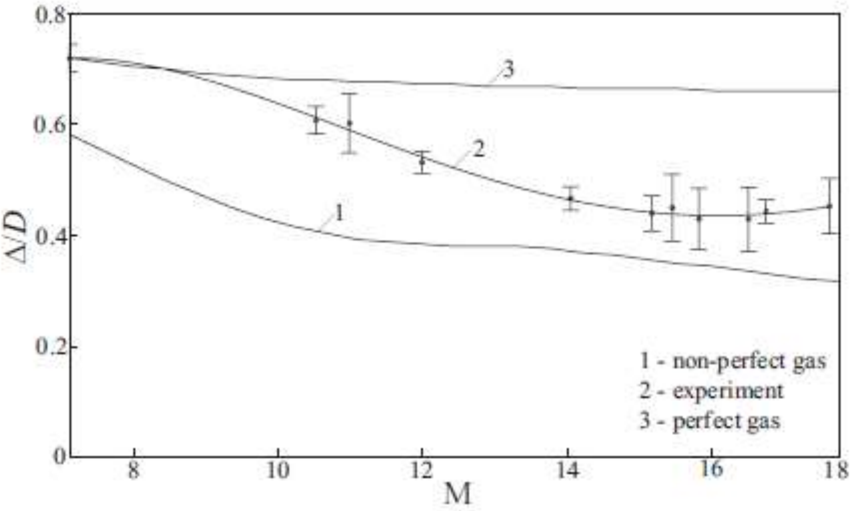

Figure 3: Dependence of the dimensionless thickness of the shock layer on Mach number

The shock standoff distance is smallest in the case of calorically perfect gas and the chemical non-equilibrium flow shock stand-off distance lies between calorically perfect gas and thermochemical equilibrium flow. In the case of calorically perfect gas, the kinetic energy of the flow ahead of the shock is mostly converted to translational and rotational molecular energy behind the shock. On the other hand, for a gas in thermal equilibrium or chemically reacting gas, the kinetic energy of the flow, when converted across the shock wave, is shared across all molecular modes of energy, and goes into the dissociation energy of the products of the chemical reactions. Hence, the temperature which is a measure of translational energy only, is less for such a case. In contrast, the pressure ratio is affected only by a small amount since pressure is a mechanically oriented variable and it is governed mainly by the fluid mechanics of the flow and not so much by the thermodynamics. The combined effect of pressure and temperature yields the density ratio across the shock which is more pronounced in case of a gas with internal energy excitation or chemically reacting gas. The shock standoff distance, which depends on the density ratio across the shock, therefore in this case is less as for non-reacting gas.

The shock standoff distance varies directly with the diameter of the sphere and inversely with density in the shock layer. For a perfect gas with $\gamma = 1.4$, the density ratio across the shock wave has a minimum value of 1/6. However, high velocity shock waves in air do not have such a limit, except at very low pressures where the relaxation times are large compared to the transit time for the air flowing over the sphere. The tendency is for the enthalpy to go into heat of formation in chemical reactions such as dissociation, rather than increasing the temperature of the air. The effect of dissociation is to increase the density and to decrease the shock detachment distance as compared to the perfect air result. For a range of Mach numbers from 4 to 10, the drag coefficient is relatively constant at a value of 0.92.

Table 4 shows the execution time for various thermodynamic air models. Time discretization is based on the third-order Runge–Kutta scheme. Flux on the faces of the control volume are calculated with Godunov and Rusanov schemes. A perfect gas model and Kraiko model are used as thermodynamic model of air.

Table 4: Time (in seconds per 1000 time steps) and speedup for flow around sphere problem.

Air model	Solver	Server 1			Server 2		
		CPU	GPU	Speedup	CPU	GPU	Speedup
Perfect gas	Godunov	2.32	0.117	19.86	4.05	0.029	139.66
	Rusanov	1.78	0.106	16.79	2.85	0.027	105.41
Kraiko model	Godunov	45.96	0.242	189.91	84.08	0.061	1378.33
	Rusanov	45.42	0.232	195.76	81.94	0.058	1412.78

6 Conclusion

The study of shock wave phenomena is of interest for the solution of problems related to the impact of shock waves on the design elements, the operation of pulsed gas dynamic devices, the use of shock waves in technological processes. Supersonic flows around wedge configurations lead to interference and diffraction phenomena complicated flow separation. These effects make complex the calculation of such phenomena using conventional numerical methods. The problem of the interpretation of the results is an important problem as well, in particular, identification of gas dynamic discontinuities.

GPUs have evolved as a new paradigm for scientific computations. Cost/performance ratio and low power consumption make GPUs attractive for high-resolution CFD computations. However, in order to exploit the inherent architecture of the device, the numerical algorithm as well as data structures are carefully tailored to minimize the memory access and any recursive relations in the computational algorithm. The finite volume method was applied to solve full Navier–Stokes equations on unstructured meshes of various topology. CUDA technology was used for programming implementation of parallel computational algorithms.

The developed CFD code is able to simulate hypersonic flows with a reasonable degree of confidence. The numerical results show that the high-temperature gas effects significantly change flow field including the standoff distance of bow shock over the front part of blunt body and other flow properties. The ability of the computations to accurately capture the shock shapes and the standoff distances demonstrates the capability of the code to model flows with high-temperature gas effects.

Acknowledgements

Authors acknowledges the support of Russian Science Foundation in the frame of Project No. 19-71-10019. This research was performed through computational resources provided by the Shared Facility Center "Data Center of FEB RAS" (Khabarovsk, Russia) and "Computer Center of SPbU" (St Petersburg, Russia).

References

1. Anderson, J. D.: Hypersonic and high temperature gas dynamics, 2nd ed. (American Institute of Aeronautics and Astronautics, Reston), p. 811, (2006)

2. Borovoi, V. Y., Skuratov, A. S., Surzhikov S. T.: AIAA Paper 2004-2634 1–11 (2004)

3. Wang, Z., Sun, X., Huang, W., Li, S., Yan, L.: Acta Astronautica 129, 95–110 (2016)

4. Yoon, S., Gnoffo, P. A., White, J. A., Thomas, J. L.: AIAA Paper 2007-4265 1–16 (2007)

5. Schmisseur, J. D.: Progress in Aerospace Sciences 12, 3–16 (2015)

6. Tchuen, G., Burtschell, Y., Zeitoun, D. E.: International Journal of Computational Fluid Dynamics 22, 1061–8562 (2008)

7. Fu, L., Gao, Z., Xu, K., Xu, A. F.: Computers and Fluids 95, 19–39 (2014)

8. Tuttafesta, M., Colonna, G., Pascazio, G.: Computer Physics Communications 184, 1497–1510 (2013)

9. Emelyanov, V. N., Karpenko, A. G., Kozelkov, A. S., Teterina, I. V., Volkov, K. N., Yalozo, A. V.: Acta Astronautica 135, 198–207 (2017)

10. Kraiko, A. N., Makarov, V. E.: High Temperature 34, 202–213 (1996)

11. Hilsenrath, J., Klein, M.: Arnold Engineering Development Center, Tullahoma, AEDC-TR-65-68 (1965)

12. Yee, H. C., Warming, R. F., Harten, A.: Lectures in Applied Mathematics 22, 357–377 (1983)

13. Lobb, R. K.: in The High Temperature Aspects of Hypersonic Flow, Vol. 68 (Elsevier, 1964), pp. 519–527