Program Complex for Fluid Dynamic Problems Simulation on GPU-Based Computer Systems

Alexander A. Davydov

Keldysh Institute of Applied Mathematics, RAS (Moscow)

ABSTRACT

This poster presents the program complex for solving CFD problems, oriented on heterogeneous GPU-based computer systems. Basing on finite volumes method the difference scheme is constructed for Quasi Gas Dynamic equations system [1, 2] in 3D formulation on arbitrary non-orthogonal structured index grid. The use of multi block grids is provided. The algorithm efficiency was verified on a set of test problems. A detailed investigation of speed-up and scaling were made. Good parallelization efficiency was achieved up to very large number of parallel GPUs in use.

NUMERICAL METHOD

QGD equation system differs from Navier–Stokes equations in some additional dissipative terms. These terms are small compared to the terms of natural viscosity and conductivity and equal to zero in flow regions where the solution satisfies the stationary Euler equations. They can be interpreted as efficient numerical stabilizers, which provide smoothness of the solution at distances of the order of mean free path.

For a 3D ideal polytrophic gas flow this system in traditional notation in the form of conservation laws approximation is

$$\frac{\partial \mathbf{U}}{\partial t} - \mathrm{Div}\mathbf{W} = 0. \qquad (1)$$

Here $\mathbf{U} = (\rho, \rho\mathbf{u}, E)^T$ – the vector of conservative variables, $E = \rho\left(\varepsilon + \mathbf{u}^2/2\right)$ — total energy, Div$\mathbf{W}$ — is the vector consisting of divergences of conservative variables fluxes $\mathbf{W}$:

$$\mathbf{W} = \begin{pmatrix} -\mathbf{j}_m \\ \Pi - \mathbf{j}_m \otimes \mathbf{u} - p\mathrm{I} \\ \Pi\mathbf{u} - \mathbf{q} - \mathbf{j}_m (E+p)/\rho \end{pmatrix}, \quad \mathrm{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \qquad (2)$$

Mass and heat fluxes, viscous stress tensor, and closing equations are defined as follows:

$$j_{mi} = \rho(u_i - w_i), \quad w_i = \frac{\tau}{\rho}\left(\frac{\partial}{\partial x_j}\rho u_j u_i + \frac{\partial}{\partial x_i} p\right), \qquad (3)$$

$$q_i = q_i^{NS} - \tau\rho u_i u_j \left(\frac{\partial}{\partial x_j}\varepsilon + p\frac{\partial}{\partial x_j}\frac{1}{\rho}\right), \qquad q_i^{NS} = -\kappa\frac{\partial}{\partial x_i}T, \qquad (4)$$

$$\Pi_{ij} = \Pi_{ij}^{NS} + \tau\rho u_i \left(u_k \frac{\partial}{\partial x_k}u_j + \frac{1}{\rho}\frac{\partial}{\partial x_j}p\right) + \tau\delta_{ij}\left(u_k\frac{\partial}{\partial x_k}p + \gamma p\frac{\partial}{\partial x_k}u_k\right), \qquad (5)$$

$$\Pi_{ij}^{NS} = \mu\left(\frac{\partial}{\partial x_i}u_j + \frac{\partial}{\partial x_j}u_i - \frac{2}{3}\delta_{ij}\frac{\partial}{\partial x_k}u_k\right), \qquad (6)$$

$$p = \rho\varepsilon(\gamma - 1), \quad T = \frac{p}{\rho R}, \quad \mu = \mu_\infty\left(\frac{T}{T_\infty}\right)^\omega, \quad \kappa = \mu\frac{\gamma R}{(\gamma-1)\,\mathrm{Pr}}, \quad \tau = \frac{\mu}{p\,\mathrm{Sc}}. \qquad (7)$$

Here γ – specific ratio, Pr and Sc – Prandtl and Schmidt numbers, τ is a relaxation parameter having a dimension of time, summation is implied over repeated indices.

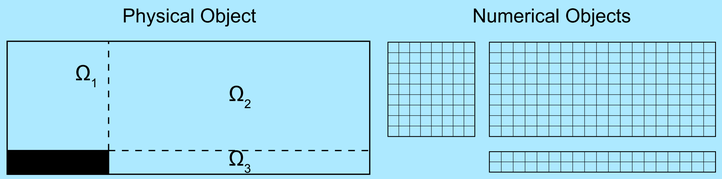
To construct difference scheme we use the control volume method. All gas dynamic parameters are addressed to cell centers. Integrating (1) over cell volume we obtain integral form of conservation laws. Replacing time derivative by finite difference, we obtain the following expression for conservative variables at the next time level:

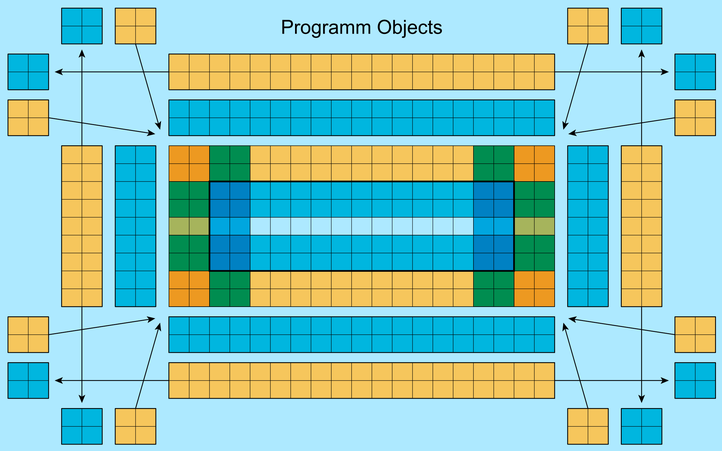
$$\hat{\mathbf{U}} = \mathbf{U} + \frac{\Delta t}{V}\sum_{l=1}^{6}\int_{S_l}(\mathbf{W}, \mathbf{n}_l)dS. \qquad (8)$$

Here summation is held over cell faces. The approximation of spatial derivatives in the fluxes of conservative variables is also based on the finite volume method. The control volumes are constructed connected with the faces of grid cells. Values of gas dynamic parameters at the centers of the cell faces are calculated using the linear interpolation.

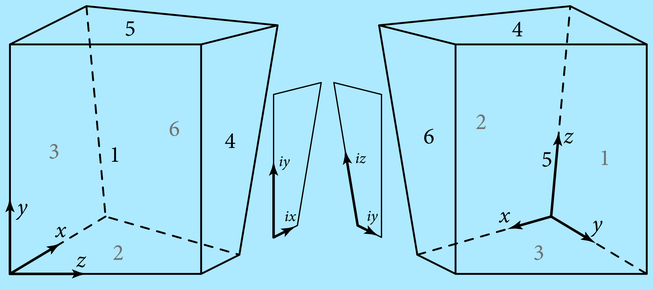
PARALLEL IMPLEMENTATION

Simulation domain divided into subdomains – blocks. Each block is processed by one device (GPU or CPU's core). Each device can process one or several blocks serially. If the block is processed on GPU, all data completely are located on GPU. And one of CPU's cores operates this GPU.

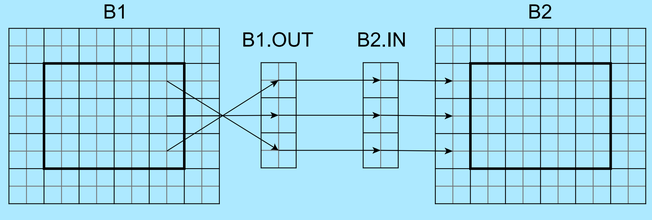




In general case, orientation of faces of the neighbor blocks doesn't coincide.



Intermediate array are used for data exchange between blocks.



PERFORMANCE

GPU/CPU speedup depending on mesh block size

Mesh block size	4050	11070	12690	19215	37515	43005	52521	101199	117547	197579
Time on GPU, s.	11	18	18	17	23	23	23	44	46	69
Time on one CPUs core, s.	29	79	96	142	283	323	407	810	978	1700
Speedup	2.6	4.4	5.3	8.4	12.3	14.0	17.7	18.4	21.3	24.6

Strong scalability

In this case mesh consist of 132 blocks with different number of cells. Total count of cells is about 2 million.

Number of GPUs (Nodes on K-100 System)	1 (1)	3 (1)	9 (3)	27 (9)
Time, s	27,90	9,88	3,58	1,35
Speedup	1,00	2,82	7,79	20,67
Efficiency, %	100,00	94,13	86,59	76,54

The results has been obtained on the K-100 computer system [3] which was constructed in KIAM RAS (Moscow, Russia) in cooperation with “KVANT” institute in 2010. K-100 is a hybrid cluster consisting of 64 nodes. Each node includes two 6-core Intel Xeon processors (2.93 GHz) and three graphical processing units NVidia Fermi C2050

Weak scalability

In this case mesh consist of 1024 blocks with 150x150x150 cells in each. Total count of cells is 3.5 billion

Number of GPUs	1	2	4	8	16	32	64	128	256	512	1024
Time, s	87,12	87,82	88,8	89,30	90,23	91,00	91,50	91,57	91,97	92,46	92,74
Efficiency %	100,0	99,2	98,1	97,6	96,6	95,7	95,2	95,1	94,7	94,2	93,9

The results has been obtained on the «Lomonosov» supercomputer system in Moscow State University [4].

Parallel CPU + GPU usage

This table illustrates advantage of paralle usagee of GPU and CPUs cores

	1 node of K-100 System			2 nodes of K-100 System		
	11 CPU	3 GPU	3GPU+8CPU	22 CPU	6GPU	6GPU+16CPU
Time, s	184.2	65.98	47.72	102	38.8	28.15
Speedup	1	2.79	3.86	1	2.62	3.62

[1] B.N. Chetverushkin. Kinetic Schemes and Quasi-Gasdynamic System of Equations, Barcelona: CIMNE, 2008.

[2] T.G. Elizarova. Quasi-Gas Dynamic Equations, Berlin Heidelberg New York: Springer-Verlag, 2009.

[3] K-100 System, Keldysh Institute of Applied Mathematics RAS, Moscow http://www.kiam.ru/MVS/resourses/k100.html

[4] «Lomonosov» Supercomputer system http://parallel.ru/cluster/lomonosov.html

This work supported by Russian Scientific Foundation. Grant № 14-11-00872