

# CFD Problems Solving Parallel Approaches on Supercomputers

Tatiana Kudryashova<sup>a</sup>, Sergey Polyakov and Viktoriia Podryga

*Keldysh Institute of Applied Mathematics RAS, Miusskaya sq. 4, Moscow 125047, Russia*

**Abstract.** The work is devoted to developing and testing parallel algorithms, suit of computer programs for numerical solution of CFD-problems on modern supercomputers. The paper summarizes our experience in solving various practical problems of gas dynamics. These problems include research of gas flow around bodies of complex shape by viscous gas flows with radiation processes near the surface of the bodies, calculations of jet streams in the open space and flows in micro channels of technical systems. We have developed approaches combining unstructured grids, domain decomposition technique, parallel implementation via MPI technology, OpenMP, CUDA. Calculations are performed on the heterogeneous computer systems with Graphical Processing Units, classical microprocessors (Central Processing Units) and use advanced parallel approaches.

## 1 Introduction

At present time there is a rapid increase in productivity of computer and supercomputer hardware. For example, some mobile or desktop systems utilizing the GPU have performance of the order of 1 teraflop and higher. Peak performance of the most powerful supercomputer system reaches about 50 PFLOPS (Guangzhou, China; 1 PFLOP means  $10^{15}$  floating point operations per second, 1 TERAFLIPS = 0,001 PFLOPS, 1000 PFLOPS = 1 EXAFLIP) [1]. Supercomputer technologies provide a progress both in the defense and civilian sectors of science and technology. Ultra-high-performance systems allow us to simulate natural and technological processes with fine detailing, simultaneous consideration of different nature factors and fast analysis of numerous possible variants. Therefore, we can get information about studied process with a high degree of accuracy, and as close as possible to real-life situations. In the field of fundamental science, the supercomputer technologies give us possibility to explore previously inaccessible for detailed modeling the phenomenon, such as problems in astrophysics, direct modeling of turbulence, genome deciphering and others.

Recently, a new methodology of interaction of supercomputer modeling techniques and experiments are formed. According to it, a detailed comparison of the calculated data and experimental data is carried out. As a result of this comparison, reliable mathematical models and computer programs are developed and in the further, they replace experiment by calculations. The further investigations are performed using computational experiment. This is a problem of storing and analyzing the huge amount of data. In this case the computational complexity of the problem increases several times (non-

linearly). The growth depends on the number of taken into account factors. These problems, for example, include an analysis of the situation in the near-Earth space (space debris), analysis of the transport situation in cities and etc.

Heterogeneous computing systems with performance about 5-10 PFLOPS widespread recently. In order to reduce the cost and power consumption, the main processing core of such systems together with conventional Central Processing Units (CPUs) constitute special calculators - Vector Processing Unit (VPU) or Graphical Processing Unit (GPU). However, combined use of CPU, GPU or VPU is complicated by a number of problems. For example, the VPU and GPU have its own memory and it differs from the CPU memory address space. Therefore, the applications must explicitly copy the data in memory VPU or GPU and back. This communication is slow and inefficient. The introduction of heterogeneous systems with coherent memory has not yet led to their mass extension and use. Thus, at present the developments of special parallel approaches to the implementation of the calculations on heterogeneous systems with different topologies and architectures are relevant [2, 3].

In this paper the Computational Fluid Dynamics (CFD) tasks on supercomputers with hybrid architecture of computing nodes, including CPU, VPU or GPU solving parallel technology is presented. Technology suggests that the discrete mathematical model of the CFD-process uses unstructured mesh and can be strongly spatially inhomogeneous. In this case, optimization techniques for balance of the computational load per a single node or a calculator inside node are implemented. To solve this problem, dynamic load balancing

<sup>a</sup> Corresponding author: kudryashova@imamod.ru

algorithms and block methods of exchange among the different devices of random access memory are proposed.

## 2 Problem definition

For investigating different CFD-problems we used system of Quasi Gas Dynamics (QGD) equations [4] and its approximations on hybrid meshes, which suppose adaptive, locally thickening meshes including both rectangular and triangular cells. The QGD system of equations is a generalization of Navier-Stokes model including additional dissipative terms with a small parameter  $\tau$  acting as an artificial regularizer.

The QGD system has a form of the conservation laws and is presented in a common form as (1-6):

$$\frac{\partial \rho}{\partial t} + \text{div} \mathbf{j}_m = 0, \quad (1)$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \text{div}(\mathbf{j}_m \otimes \mathbf{u}) + \nabla p = \text{div} \mathbf{\Pi} + \mathbf{F}, \quad (2)$$

$$\frac{\partial}{\partial t} \left[ \rho \left( \frac{\mathbf{u}^2}{2} + \varepsilon \right) \right] + \text{div} \left[ \mathbf{j}_m \left( \frac{\mathbf{u}^2}{2} + \varepsilon \right) \right] = \text{div}(\mathbf{\Pi} \otimes \mathbf{u}) - \text{div} \mathbf{q} + Q \quad (3)$$

Mass flux density vector is expressed as:

$$\mathbf{j}_m = \rho \mathbf{u} - \tau (\text{div}(\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p), \quad (4)$$

shear stress tensor has the following form

$$\begin{aligned} \mathbf{\Pi} = & -\tau \text{div}(\rho \mathbf{u} \otimes \mathbf{u}) \otimes \mathbf{u} - \tau (\nabla p \otimes \mathbf{u}) + \\ & + \tau \left[ \text{div}(\rho \mathbf{u} \otimes \mathbf{u} \otimes \mathbf{u}) + (\nabla \otimes p \mathbf{u}) + (\nabla p \mathbf{u})^T \right] + \\ & + \mathbf{I} \tau \text{div}(p \mathbf{u}), \end{aligned} \quad (5)$$

heat flux vector:

$$\mathbf{q} = -\chi \nabla T - \tau p \nabla \left[ \frac{p}{\rho} + \varepsilon \right] - \tau p \mathbf{u} \left[ (\mathbf{u} \cdot \nabla) \varepsilon + p (\mathbf{u} \cdot \nabla) \left( \frac{1}{\rho} \right) \right] \quad (6)$$

Here

$E = \rho |\mathbf{u}|^2 / 2 + p\varepsilon$ ,  $\varepsilon = c_v T$ ,  $p = Z\rho(R/M)T$ ,  $\tau = \mu / p$ ,  $\mu$  is the viscosity coefficient,  $M$  is the molar mass,  $R$  is the universal gas constant,  $\otimes$  – a sign of inner product and  $\mathbf{I}$  – unity matrix.  $\mathbf{F}$  – momentum vector of external forces,  $Q$  – power density of external energy sources,  $\rho$  – density,  $\mathbf{u}$  – vector of velocity,  $Z$  – compressibility factor,  $c_v$  – coefficient of heat capacity at constant volume,  $\chi$  – coefficient of thermal conductivity,  $T$  – temperature.

System (1)–(6) is completed by the equations of state for gas, suitable boundary conditions and the expressions for the coefficients of viscosity, heat conductivity and  $\tau$  coefficient.

For special problems QGD system is supplemented by additional terms. In this paper we consider two applied tasks: radiation transfer and gas mixture flows.

## 2.1 Radiative Gas Dynamics

Many space programs are carried out today. They include launches of interplanetary stations (ExoMars 2016) and International Space Station flights (MKS, Soyuz Flight VS14) [5]. Therefore, scientific explorations of this area are quite relevant.

Returning to Earth after a flight into space is a fundamental problem, that demands to take into account gas-dynamic flow processes and the effects associated with radiation. Investigations of flows around a re-entry vehicle entering the atmosphere are traditional for high temperature gas dynamics because superheating proved a serious problem.

Development of space vehicles thermal insulation is important aspect of designing spacecraft. It requires accurate prediction of environmental gas parameters. An accurate description of the aero thermal environment is required to minimize the weight and price of the Thermal Protection System. The inability of test facilities to reproduce the high enthalpy flow coupled with the prohibitive expense of flight tests leads to the use of an analytical method, namely CFD, to describe the flow.

Radiative Gas Dynamics (RGD) approach is implemented for computing. The model includes QGD equations system and radiation transfer equations. The radiation transfer is realized with the diffusion approximation model [6-9]. The diffusion approximation model was coupled with the gas dynamic model by including the radiation energy  $Q = \text{div} \vec{W}$  in the energy equation (3', 7, 8):

$$\begin{aligned} \frac{\partial}{\partial t} \left[ \rho \left( \frac{\mathbf{u}^2}{2} + \varepsilon \right) \right] + \text{div} \left[ \mathbf{j}_m \left( \frac{\mathbf{u}^2}{2} + \varepsilon \right) \right] = \\ = \text{div}(\mathbf{\Pi} \otimes \mathbf{u}) - \text{div} \mathbf{q} - \text{div} \mathbf{W}, \end{aligned} \quad (3')$$

where

$$\mathbf{W} = \int_0^\infty d\nu \int_{4\pi} I_\nu \mathbf{\Omega} d\mathbf{\Omega}, \quad (7)$$

$$\mathbf{\Omega} \nabla I_\nu = \chi_\nu (I_{\nu p} - I_\nu), \quad (8)$$

where  $\nu$  is photon frequency,  $\mathbf{\Omega}$  – the direction of photon propagation,  $I_\nu$  – radiation intensity, corresponding to a specific frequency,  $I_{\nu p}$  – the intensity of the equilibrium radiation,  $\chi_\nu$  – absorption coefficient.

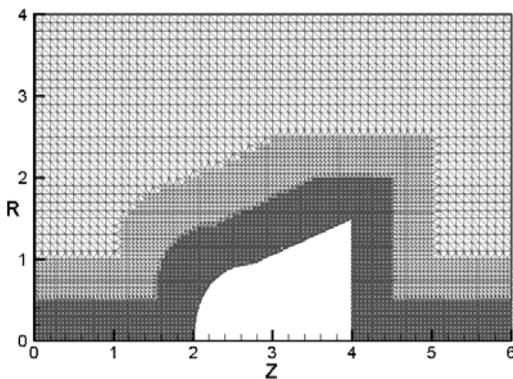
The diffusion equation (8) can correspond either individual spectral lines or a number of frequency intervals [6, 7]. We use the second approach and set 600 frequency groups for air. Tabulated data from [8] have been used for our early computations [9, 10]. Database supports approximately  $10^5$  meanings of the absorption coefficients. The four species accounted in these calculations are  $O_2, N_2, NO, N_2^+$ . Initial conditions of the free stream for calculations are presented in Table 1.

For realization of our mathematical approach the special parallel software was used [11-14].

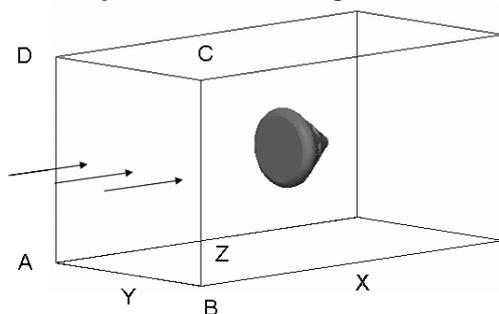
**Table 1.** The initial conditions of the free stream.

Background gas temperature $T_0(K)$	266
Temperature of head shield $T_1(K)$	2000÷3300
Pressure $p_0(Pa)$	4.3*10
Density $\rho_0(kg/m^3)$	5.63*10 <sup>-4</sup>
Characteristic size of problem $L(m)$	5.035*10 <sup>-1</sup>
Mach number $M_\infty$	12; 24

Gas flow is investigated inside free form closed volume. The boundaries are made of triangles or rectangles in 2D and 3D space. The unstructured locally-condensing meshes are applied. The number of nodes in the used meshes varies up to 10<sup>6</sup> in 2D case and more 10<sup>6</sup> in 3D case [9, 10]. Computational domains for 2D and 3D cases are shown in Figure 1, 2.



**Figure 1.** Computational domain with grid for 2D variant.



**Figure 2.** Computational domain.

For QGD equations a time explicit conservative numerical scheme was developed. Explicit schemes are very convenient for parallel computing. The main technique of parallelization is domain decomposition. Inter-communications among cluster nodes are performed with MPI library and OpenMP technology. For RGD calculations we realized parallelizing by groups: linear systems of each spectral group are computed by their own CPU cores. Then the sum of independent results is calculated. Advantages of this approach: simple realization, high computation speed, free selection of linear system solution method and pre-conditioner. Disadvantages are: inability to utilize more CPU cores than spectral group count (600), absence of load

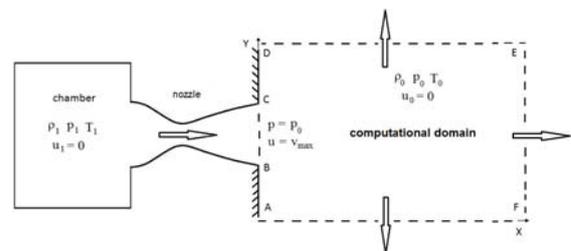
balancing. Other variant is the use of parallel algorithms for solving linear equations systems. This allows us to increase number of CPU cores but efficiency of these methods significantly decreases when the number of cores grows.

General parallel algorithm integrates the advantages of previous two methods. All available CPU cores are divided into groups and each group processes its own set of linear equations systems. Parallel algorithm for linear equations systems solving is used within each group and all the data are stored distributively.

## 2.2 Gas mixture flows

Another problem in which we used a developed parallel approach is related to modeling flows of gas mixtures in microchannels technical systems. Here we present the study of the evolution of a binary supersonic gas jet that is propagated in a vacuum. The computing complexity of such flows is associated with the violation in some parts of the computational domain of continuity hypothesis.

Using a multiscale approach is a way to overcome the problem of discontinuity [15-20]. We propose method that integrates a macroscopic approach and a method of molecular dynamics (MMD) [21-23]. Macroscopic approach is based on the QGD system [4]. The flow parameters are corrected with MMD. The system of molecular dynamic equations is used as a subgrid algorithm. At first step for MMD we assume that the molecules are distributed uniformly in space. Velocities of the molecules are distributed according to the Maxwell equilibrium function. Computational domain is shown in Figure 3.



**Figure 3.** Computational domain.

The QGD equations of binary gas mixture dynamics are written as following (1)-(3), where F and Q are formulas (9, 10):

$$\mathbf{F}_a \equiv \mathbf{S}_a^u = v_{ab} \rho_a (\mathbf{u}_a^* - \mathbf{u}_a), \quad \mathbf{F}_b \equiv \mathbf{S}_b^u = v_{ba} \rho_b (\mathbf{u}_b^* - \mathbf{u}_b); \quad (9)$$

$$Q_a \equiv S_a^E = v_{ab} (E_a^* - E_a), \quad Q_b \equiv S_b^E = v_{ba} (E_b^* - E_b). \quad (10)$$

A mixture consists of two gases: *a* and *b* with numerical densities  $n_a$  and  $n_b$  and mass densities  $\rho_a = m_a n_a$  and  $\rho_b = m_b n_b$ , respectively, here  $m_a$  and  $m_b$  are the masses of molecules of the gases *a* and *b*. Each of the gases is characterized by its temperature  $T_i$  and the macroscopic velocity –  $\mathbf{u}_i$ . The gas constant is equal to  $\mathfrak{R}_i = k_B / m_i$ , where  $k_B$  is the Boltzmann constant,  $\gamma_i$  is the specific heat ratio,  $Pr$  is the Prandtl number,  $\tau$  is a relaxation parameter containing the dimension of time.

Here  $S_a^u, S_b^u, S_a^E, S_b^E$  are the exchange terms (5)-(6).  $\mathbf{u}_l^*, E_l^*$  – vector of velocity and the total energy density, allowing us to take into account redistribution of momentum and energy between the components of the mixture. These parameters will be determined through MMD below.  $\nu_a$  and  $\nu_b$  – the frequency of mutual collisions between the molecules of one kind,  $\nu_{ab}$  – the frequency of collisions between the molecules of  $a$  kind and the molecules of  $b$  kind,  $\nu_{ba}$  – the frequency of collisions between the molecules of  $b$  kind and the molecules of  $a$  kind.

The calculation gas parameters are shown in the Table 2.

**Table 2.** The gas parameters.

Diameter (BC), $\mu m$	Pressure into chamber $p_1, Pa$	Temperature into chamber, $T_1, K$	Pressure into domain $p_0, Pa$	Temperature into domain $T_0, K$
6÷300	$10^3$	273	$10^3$	273

The parallel algorithm is based on "domain decomposition" technique and on splitting into physical processes. Splitting on physical processes means that QGD equations are processed by CPUs and MD is processed by CPUs, VPUs or GPUs. Gas dynamics is calculated on CPU units. For MMD parallelization an equal number of particles is sent to each computing unit. According to the algorithm the molecular dynamic calculations do not require exchanges.

Domain decomposition on computational domain is used for QGD equations. We take into account four levels on parallelization: nodes of system → CPUs → CPU cores → CPU threads. Node of system and GPU or VPU get the same subdomains. GPU or VPU computes the exchange terms. If GPU or VPU is unavailable these calculations are produced by CPU threads.

Parallelization of MD calculations is produced by splitting the set of molecules into groups with the same power. Each GPU or VPU produces a few molecular groups.

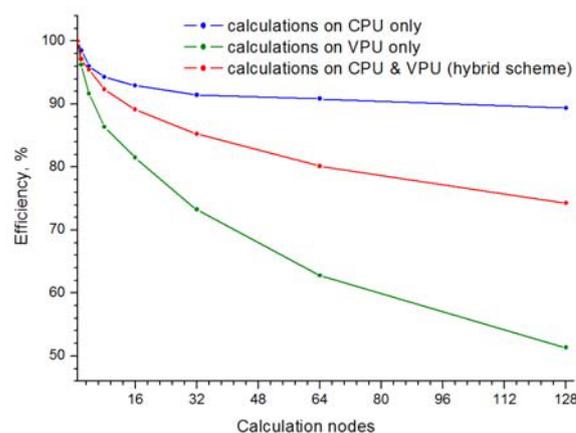
### 3 Results

For solution of described problems, we constructed the parallel software suite that is based on hybrid technology. The technology combines MPI, OpenMP and CUDA. The parallel testing techniques developed by us were performed using various computing systems. For demonstration of hybrid calculations, two systems were implemented: the K-100 hybrid cluster with CPU and GPU set at Keldysh Institute of Applied Mathematics RAS (Moscow), the MVS-10P supercomputer at Joint Super Computer Center of RAS based on CPUs and VPUs. The results of our studies for each of these tasks are presented below.

### 3.1. Radiative Gas Dynamic computing

Early for solution of RGD problem we used two parallel systems SKIF MSU and MVS-100k. Group parallelizing efficiency using 600 CPU cores was 84%. Hybrid parallelizing (groups consisting of 8 CPU cores) reached 79% efficiency using 600 CPU cores and 66% efficiency using 1200 cores.

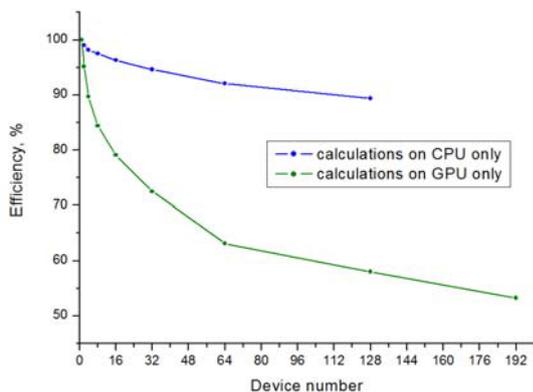
Now we use the PETAFLOPS-system MVS-10P with CPU and VPU devices. The system has 207 nodes. Each node has two CPU Intel Xeon E5-2690 (8 cores and 16 threads per one CPU) and two VPU Intel Xeon Phi 7110X (41 cores and 244 threads per one VPU). Parallelization scheme was based on domain decomposition technique. Program realization was based on MPI and OpenMP technologies. For parallelization on spectral groups we used threads of CPU or VPU. The results of testing of our parallel software on space 3D grid with  $48 \cdot 10^6$  hybrid cells are shown in Figure 4. These results confirm that the CPU calculations are more effective. However the using of all devices on each node allows minimizing the computation time. And as summary, high possibilities of modern hybrid computer systems allows solving large RGD-problems.



**Figure 4:** Efficiency of RGD algorithm parallelized with the help of hybrid approach (on group and on space).

### 3.2 Calculation of gas mixture flows

Calculations of gas mixture flows were provided on K100 hybrid cluster with CPU and GPU. Each node of the cluster has two CPU Intel Xeon X (6 cores and 12 threads per one CPU) and three GPU NVidia Tesla C2050 (448 CUDA cores per one GPU). We tested of our software on 3D Cartesian grid with sizes  $240 \times 240 \times 720$  cells. The calculations were connected with evolution of nitrogen and hydrogen jet in nickel microchannel with sizes  $30 \times 30 \times 90$  microns. Each grid cell in dependence on physical conditions can consist of 500 - 50000 gas particles. The results of testing are shown in Figure 5. These results confirm also that using of CPU systems is more effective. Perhaps the using of GPU allows reducing calculation time by 6 - 10 times. The using of more modern GPU devices (NVidia Tesla K40 or K80) allows decreasing calculation times by 20-40 times [24].



**Figure 5:** Efficiency of QGD-MD algorithm obtained by separate calculations on CPUs and GPUs.

## 4 Conclusions

Supercomputing can accelerate scientific research in important areas such as physics, material sciences, biology, and etc. Supercomputer simulations can augment or replace experimentation in cases where experiments are hazardous, expensive, or even impossible to perform. High performance systems allow us to solve large problems in the reasonable time.

They provide great computing power and possibility. To use them we need special tools and approaches. In our work we presented parallel algorithms and realization methods of the algorithms on high performance computer systems with different processors. Numerical results and data of efficiency reflect the correct direction of researches.

As computational power increases, fewer compromises are made, and more accurate results can be obtained. Therefore, in many application domains, there is essentially no limit to the amount of compute power and many contemporary challenges can be overcome.

## Acknowledgements

This work was supported by Russian Foundation for Basic Research (grants № 15-29-07090-ofi\_m and 16-07-00206-a).

## References

1. <http://www.top500.org/list/2015/06/>
2. <http://www.parallel.ru>
3. <https://www.ot.ru/en/solutions/hybrid-supercomputer/>
4. T.G. Elizarova, *Quasi-Gas Dynamic Equations* (Springer-Verlag, Berlin Heidelberg New York, 2009)
5. <http://www.arianespace.com/mission/ariane-flight-vs14>
6. B.N. Chetverushkin, *Mathematical modeling of problems in the dynamics of a radiating gas* (Science, Moscow, 1985) [In Russian]

7. T.K. Bose, *High Temperature Gas Dynamics* (Springer, Cham, Heidelberg, New York, Dordrecht, London, 2014)
8. I.V. Avilova, L.M. Biberman, et al., *Optical Properties of Hot Air* (Science, Moscow, 1970) [In Russian]
9. T.A. Kudryashova, S.V. Polyakov, A.A. Sverdlin, *Math. Models Comput. Simul.*, **1**(4), 445–452 (2009)
10. T.A. Kudryashova, S.V. Polyakov, E.M. Kononov, *Math. Models and Comput. Simul.*, **1**(5), 551–560 (2009)
11. METIS-Family of Multilevel Partitioning Algorithms, <http://glaros.dtc.umn.edu/gkhome/views/metis> (2009)
12. S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M.G. Knepley, L. Curfman McInnes, B.F. Smith, H. Zhang, <http://www.mcs.anl.gov/petsc> (2001)
13. H. Si, *TetGen: A Quality Tetrahedral Mesh Generator*, <http://tetgen.berlios.de/> (2009)
14. K. Lipnikov, Yu. Vassilevski, *Engineering with Computers*, **20**(3), 193–201 (2004)
15. V.A. Titarev, *Comput. Math. Math. Phys.*, **49**(7), 1197–1211 (2009)
16. M.K. Borg, D.A. Lockerby, J.M. Reese, *J. Fluid Mech.*, **768**, 388–414 (2015)
17. K. Morinishi, *Computers and Fluids*, **35**(8-9), 978–985 (2006)
18. A. Patronis, D.A. Lockerby. *J. Comput. Phys.*, **270**, 532–543 (2014)
19. S.Y. Docherty, M.K. Borg, D.A. Lockerby, J.M. Reese, *Int. J. Heat and Fluid Flow*, **50**, 114–125 (2014)
20. A. Alexiadis, D.A. Lockerby, M.K. Borg, J.M. Reese, *Comput. Methods Appl. Mech. Eng.*, **264**, 81–94 (2013)
21. Yu.N. Karamzin, T.A. Kudryashova, V.O. Podryga, S.V. Polyakov, *Math. Model.*, **27**(7), 65–74 (2015) [In Russian]
22. Yu. Karamzin, T. Kudryashova, V. Podryga, S. Polyakov, *Int. Conf. Eng. Comput. Tech. (ECT 2014)*, **28**, (Civil-Comp Press, Stirlingshire, Scotland, 2014)
23. T. Kudryashova, Yu. Karamzin, V. Podryga, S. Polyakov, *Adv. Eng. Software* (to be published)
24. <http://www.nvidia.co.uk/object/k80-gpu-test-drive-uk.html>