ALGORITHMS FOR THE CALCULATION OF NONLINEAR PROCESSES ON HYBRID ARCHITECTURE CLUSTERS

A.V. Bogdanov [a], V.V. Mareev [b] and N. Storublevtcev [c]

St.Petersburg State University, 7/9 Universitetskaya nab., St. Petersburg, 199034 Russia

E-mail: [a] bogdanov@csa.ru, [b] map@csa.ru, [c] 100.rub@mail.ru

The problem of porting programs from one hardware platform to another has not ceased to be less relevant and simpler with time. The purpose of our work is to identify the key features of algorithms in porting codes for calculating of essentially nonlinear processes to a modern cluster of hybrid architecture that includes both CPUs (Intel Xeon) and GPU (NVIDIA TESLA) processors. As a test problem for studying the process of porting a code to a cluster of hybrid architecture, the KPI equation of Kadomtsev-Petviashvili was chosen, written in integro-differential form [1], [2].

Keywords: High performance computing, CPU architectures, GPU, FPGA

1. Introduction

Porting a computational task to a graphics processor is a difficult problem. As a rule, the program is ported to a graphics accelerator for the sake of improving performance. The main problem during the transfer is to preserve the correctness of program execution.

It is impossible to transfer the entire code to the graphics processor. The code will always be launched from the central processor. In any program, there are serial sections of code that cannot be parallelized and thus are meaninglessly transferred to a graphics processor due to its peculiarities, nature of GPU architecture and the increased cost of data transfer.

2. The test problem

As a test problem consider the two-dimensional Kadomtsev-Petviashvili equation — KPI

$$[u_t + 0.5(u^2)_x + \beta u_{xxx} - G]_x = \eta u_{yy} \tag{1}$$

Equation (1) with respect to function $u(x,y,t)$ is considered in the domain $t \geq 0$, $x, y \in (-\infty, \infty)$, $\beta, \eta \geq 0$, $G(x,y)$ is external source [1].

Instead of the original equation (1) its integro-differential analogue is considered [2]

$$u_t + 0.5(u^2)_x + \beta u_{xxx} = \eta \int_{-\infty}^{x} u_{yy}\,(x', y, t)dx' + G(x,y) \tag{2}$$

Solution of the equation (2) in half-plane $t \geq 0$ is sought for initial distribution $u(x,y,0) = q(x,y)$. The numerical simulation of the equation (2) is carried out using a linearized implicit finite-difference scheme using in some cases the flux correction procedure (FCT) [3].

For equation (2), the approximation is performed using the central-difference operators.

$$\begin{aligned} u_{j,k}^{n+1} - u_{j,k}^{n} + \frac{\Delta t}{4\Delta x}\left(F_{j+1,k}^{n+1} - F_{j-1,k}^{n+1}\right) + \beta\frac{\Delta t}{2\Delta x^3}\left(u_{j+2,k}^{n+1} - 2u_{j+1,k}^{n+1} + 2u_{j-1,k}^{n+1} - u_{j-2,k}^{n+1}\right) \\ = \Delta t \eta S_{j,k}^{n+1} + \Delta t G_{j,k} \end{aligned} \tag{3}$$

The resulting system of difference equations (3) is reduced to the form:

$$a_j \Delta u_{j-2,k}^{n+1} + b_j \Delta u_{j-1,k}^{n+1} + c_j \Delta u_{j,k}^{n+1} + d_j \Delta u_{j+1,k}^{n+1} + e_j \Delta u_{j+2,k}^{n+1} = f_{j,k}^{n} \tag{4}$$

with $\Delta u_{j,k}^{n+1} = u_{j,k}^{n+1} - u_{j,k}^{n}$ and $F_{j,k}^{n+1} \equiv (u^2)_{j,k}^{n+1} = (u^2)_{j,k}^{n} + 2u_{j,k}^{n}\Delta u_{j,k}^{n+1} + O(\Delta t^2)$

Notations used in equation (3) traditional for finite difference schemes:

$$u(j\Delta x, k\Delta y, n\Delta t) = u_{j,k}^{n}, \quad \int_{-\infty}^{x_j} u_{yy}dx' \approx \int_{x_{\min}}^{x_j} u_{yy}dx' \equiv S_{j,k}^{n},$$

with $\Delta x, \Delta y$ being the spatial coordinates steps, Δt being the time step, $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \times [0, T]$ — computational domain.

The boundary conditions are used: $u_x = u_{xx} = 0$ along boundary lines x_1 and x_M, and $u_y = 0$ along the lines y_1 and y_L ($x_{\min} = x_1$, $x_{\max} = x_M$, $y_{\min} = y_1$, $y_{\max} = y_L$:

$$u_{-1,k}^{n} = u_{0,k}^{n} = u_{1,k}^{n}; \;\; u_{M+2,k}^{n} = u_{M+1,k}^{n} = u_{M,k}^{n}; \;\; u_{j,0}^{n} = u_{j,1}^{n}; \;\; u_{j,L+1}^{n} = u_{j,L}^{n}$$

The system (4) is solved by a five-point run.

As an initial distribution is considered the ellipsoid of rotation:

$$q(x,y) = c_1\sqrt{1 - \frac{x^2}{a_1^2} - \frac{y^2}{b_1^2}}, \tag{5}$$

with the volume $V_1 = 2\pi a_1 b_1 c_1/3$, and a_1, b_1, c_1 being the half axis.

Similarly to the initial distribution (5), the distribution of sources as an ellipsoid of rotation is chosen:

$$G(x,y) = c_2\sqrt{1 - \frac{(x-x_0)^2}{a_2^2} - \frac{(y-y_0)^2}{b_2^2}}$$

with the volume $V_2 = 2\pi a_2 b_2 c_2/3$, and a_2, b_2, c_2 being the half axis, (x_0, y_0) — center of ellipsoid.

The proposed approach is quite natural for porting to GPGPU since it consists of many iterations within which it is necessary to solve large systems of linear equations. Taking into account peculiarities of GPGPU architecture [4] we solve systems of linear equations on GPGPU leaving all pre- and postprocessing to CPU. This approach is realized by semi-automatic procedure, described in [4].

In Figure 1, 2 we show the moments of the perturbations evolution for the values $a_1 = 2$, $b_1 = 3$, $c_1 = 7.5$, т.е. $V_1 = 20\pi$ and $a_2 = 2$, $b_2 = 3$, $c_2 = 2.5, x_0 = -14, y_0 = 14, V_2 = 10\pi$. The calculation was carried out without FCT procedure.

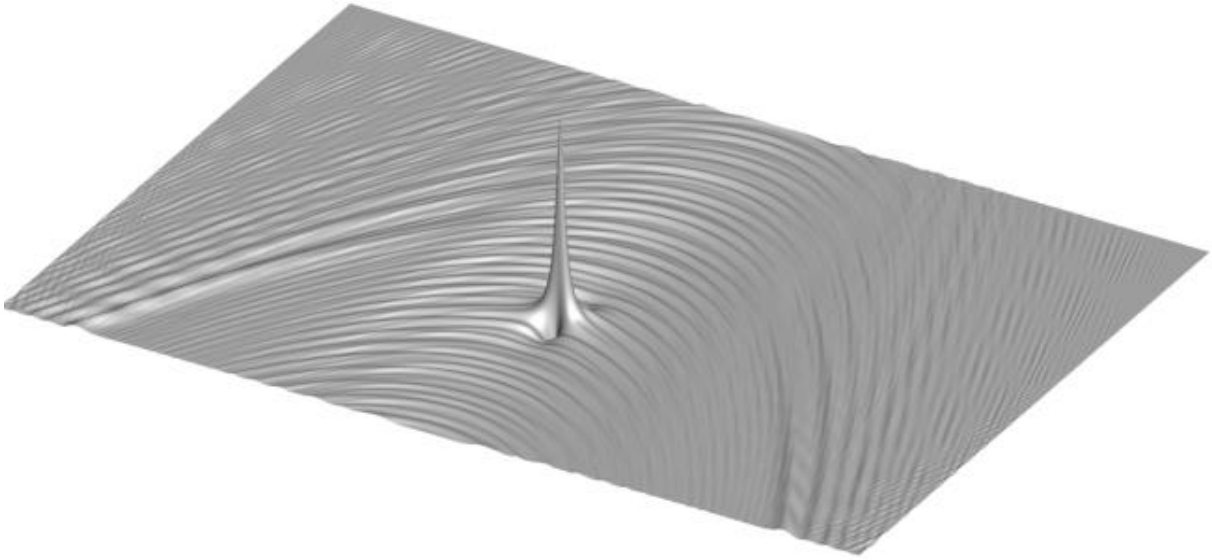

Figure 1. Without source at $t = 16.5$. Grid: 600×850, $\Delta t = 10^{-4}$, $\Delta x = \Delta y = 0.2$

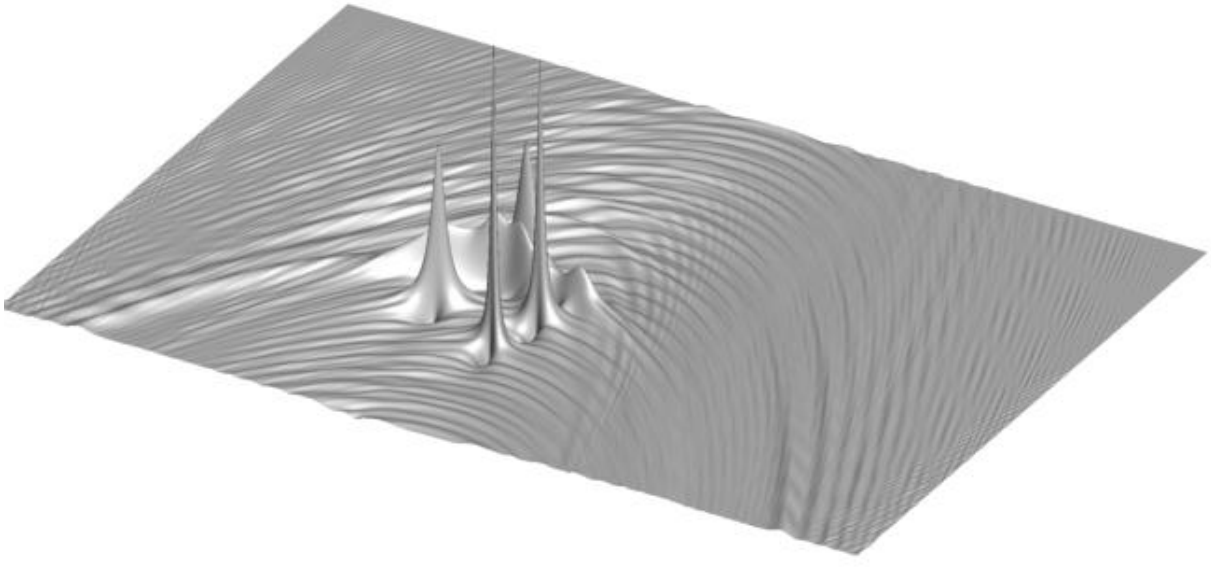

Figure 2. 3D perturbation with source at $t = 16.5$. Grid: 600×850, $\Delta t = 10^{-4}$, $\Delta x = \Delta y = 0.2$

3. Conclusion

1. As a result of our approach, an algorithm was proposed for transferring the simulation program for a two-dimensional nonstationary model problem to a hybrid system. The features of such a transition are revealed.
2. The use of modern hybrid systems in combination with the new algorithmic approach has allowed also to create a software and hardware platform for mass computations of wave processes.
3. There are no substantial bottlenecks for GPGPU onboard memory and the attempts to use heterogeneous systems for 3D computations are justified.

Acknowledgement

This work was supported by the grant of Saint Petersburg State University no. 26520170 and the Russian Foundation for Basic Research (RFBR), grant #16-07-01113.

References

[1] Bogdanov A.V., Mareev V.V. Numerical Simulation KPI Equation // Proceedings of the 15th International Ship Stability Workshop, June 2016, Stockholm, Sweden. pp. 115-117.

[2] Bogdanov A., Mareev V., Kulabukhova N., Shchegoleva N. Influence of External Source on KPI Equation. Lecture Notes in Computer Science book series (LNCS, volume 10963), 2018, pp 123-135

[3] Fletcher C.A.J. Computational Techniques for Fluid Dynamics 1 // 2nd edition. – Springer-Verlag, 1991. 401 p.

[4] Alexander Bogdanov, Nikita Storublevtcev, Vladimir Mareev. On porting of applications to new heterogeneous systems // Proceedings of the VIII International Conference "Distributed Computing and Grid-technologies in Science and Education" (GRID 2018) [In print]