УДК:614.8

MODELING OF PHYSICAL AND CHEMICAL PROCESSES USING CUDA TECHNOLOGY

A. Kabdoldina, Ye. Tuleshov, Zh. Ualiyev
Institute of Combustion Problems

Abstract
Application of numerical methods is getting more and increasingly popular in relation to growing labor-intensive experiments and the complexity of obtaining detailed information about the flows and chemical processes that occur in them. For example: in the study of chemical lasers, modeling the processes occurring in a fire. This paper describes the development of a system of fast CUDA parallel algorithms for solving difference equations (Navier - Stokes, Poisson) and chemical kinetics, obtained in problems with the modeling of reactions flowing in their variety of substances.

Key words: modeling, fire, information, chemical flows

Introduction

Computational methods for solving the problems of physical and chemical processes require large computer resources. Thus, the parallelization on supercomputers with GPU is the actual direction of computing technology. Simulation of flow with the complicated chemical processes is a complex mathematical and computational problem. In most cases, the Navier - Stokes equations that describe the behavior of a viscous incompressible fluid are approximated by various grid methods. The basic requirement for such methods is to provide high accuracy of the results with the minimum necessary computational cost (time, memory).

In solving the Navier - Stokes equations is required repurposed the Poisson equation. It is computationally complex and is the main computational load.

Simulation active reactive environment requires reliable methods for solving stiff differential equations. As such a method may be a Gear's method based on backward differentiation formulas. Its advantage over the multi-step method is the ability to use strategies with variable step of integration and simple construction of the parallel circuit. In the field of scientific calculations become much more popular computing using graphics accelerators.

Modern graphics accelerators are massively parallel processors with shared memory. Unlike central processor with multiple cores, a graphics processor contains several tens to several thousands of cores that can perform computing in parallel. The most advanced technology at the moment is a system of CUDA (Compute Unified Device Architecture).

Formulation of the problem

Theoretical Hydrodynamics tends approximately predict the flow of a real fluid by solving boundary problems for the relevant systems of differential equations in partial derivatives. Numerical study of convective flows, the development of technologies for their mathematical modeling besides applied aspect are fundamental. Vortex structure of fluids and gases, which result from the interaction of hydro-mechanical fields with thermal or electrical, are highly organized spatial structures in dissipative media far from thermodynamic equilibrium. They are characterized by a complex spatial-temporal dynamics, the plurality of regimes, the effects of turbulence and others.

For modeling thermal convective and hydro mechanical processes are used the Navier-Stokes equations in the Boussinesq approximation [1]:

$$0\frac{\partial \vec{V}}{\partial t} + (\vec{V}\nabla)\vec{V} = -\frac{1}{\rho_0}\nabla p' + \nu\Delta\vec{V} + \beta\theta\vec{g}\vec{n}, \quad (1)$$

$$\frac{\partial \theta}{\partial t} + (\vec{V}\nabla)\theta = a\Delta\vec{V}, \quad (2)$$

$$div\,\vec{V} = 0, \quad (3)$$

where $\vec{V}$ - the velocity vector;
ν - the kinematic viscosity;
β - coefficient of thermal expansion;
g - the gravitational acceleration vector;
n - unit outward normal vector;
a - thermal diffusivity; the temperature θ is measured from the average of a constant value

corresponding to the density ρ_0;

p' is the pressure deviation from the hydrostatic pressure corresponding to a constant average temperature and density ρ_0.

Methods for solving the Navier - Stokes equations in vortex - stream function perfectly suited for many two-dimensional problems. Use the following two-dimensional modification of equations (1) - (3) in the variables "vorticity - stream function - temperature", allowing to reduce the number of equations and eliminate the pressure:

$$\frac{\partial\omega}{\partial t}+u\frac{\partial\omega}{\partial x}+\upsilon\frac{\partial\omega}{\partial y}=\left(\nu_{xx}\frac{\partial^2\omega}{\partial x^2}+\nu_{yy}\frac{\partial^2\omega}{\partial y^2}\right)+g_x\beta\frac{\partial\theta}{\partial y}-g_y\beta\frac{\partial\theta}{\partial x}, \tag{4}$$

$$\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial y^2}=\omega, \tag{5}$$

$$\frac{\partial\theta}{\partial t}+u\frac{\partial\theta}{\partial x}+\upsilon\frac{\partial\theta}{\partial y}=\left(a_{xx}\frac{\partial^2\theta}{\partial x^2}+a_{yy}\frac{\partial^2\theta}{\partial y^2}\right)+b\theta+Q. \tag{6}$$

Here $\omega=rot\vec{V}$ - vorticity - vector quantity describing the local rotation of the fluid;

ψ - stream function related with the components of the velocity vector expressions $u = \partial\psi / \partial y$, $\upsilon = -\partial\psi / \partial x$.

Unlike classical model of the Boussinesq equation (4) - (6) allow to describe the flow of continuous media with both isotropic and with anisotropic properties. Developed turbulent flows ν_{xx}, ν_{yy}, a_{xx}, a_{yy} - coefficients of turbulent viscosity and thermal diffusivity. For the case of nematic fluids ν_{xx}, ν_{yy} - combination of Leslie coefficients, a_{xx}, a_{yy} - the principal values of the tensor diffusivity along and perpendicular to the nematic axis. Furthermore, in equation (6) added to distributed internal heat sources, depending on the temperature, where Q - capacity of indoor sources, b - the coefficient of nonlinearity.

To solve the problem of chemical hydrodynamics, the study of the process of transfer of matter in chemical reactions, it is necessary to add a mass transfer equation. Transport of material in a moving medium is caused by two different mechanisms. Availability the difference in concentrations in the fluid (gas) causing the molecular diffusion, besides, the particles of the substance dissolved in the fluid are entrained by the latter during its motion and carried along with it. The set of the two processes is called convective diffusion:

$$\frac{\partial C_j}{\partial t}+\sum_{i=1}^{2}U_i\ \frac{\partial C_j}{\partial x_i}=D_{C_j}\sum_{i=1}^{2}\frac{\partial^2 C_j}{\partial x_i^2}+W_j;$$

$$j=\overline{1,N}, \tag{7}$$

where C_j - the concentration of the j-th substance;

D_{C_j} - diffusion coefficient of the j-th substance;

N - number of substances.

Member of W_j determines the change in the concentration of the j-th substance as a result of chemical reactions over time.

The last member of the convective diffusion equation described by the equations of chemical kinetics, which studies the chemical reaction as a process that occurs over time, the mechanism of this process and its dependence on the conditions of implementation. Chemical kinetics establishes temporal regularities of chemical reactions, the relationship between the reaction rate and the terms of its implementation, identify factors that affect the speed and direction of chemical reactions.

In the simulation of chemical reactions we assume that there are N substances involved in the reaction, then the system of equations will have the form:

$$\sum_{i=1}^{N}L_{ij}P_i\ \rightarrow\ \sum_{i=1}^{N}R_{ij}P_i;\ j=\overline{1,K}, \tag{8}$$

where P_i - i-th substance;

L_{ij} - stoichiometric coefficient of the i-th material to the left side of the j-th reaction;

R_{ij} - stoichiometric coefficient of the i-th material to the right side of the j-th reaction. Matrix **L** = [L_{ij}] and **R** = [R_{ij}] have dimension (N x K).

The system of differential equations of chemical kinetics is determined by a set of chemical reactions. It is a system of ordinary differential equations of the first order to the following:

$$W_i = \frac{dC_i}{dt} == \sum_{j=1}^{K} R_{ij} A_j \prod_{m=1}^{N} C_m^j - \sum_{j=1}^{K} L_{ij} A_j \prod_{m=1}^{N} C_m^j,$$

$$i = \overline{1, N}, \tag{9}$$

wherein A_j- rate constant j-th reaction.

In considering the equilibrium and quasi-equilibrium process is limited to assess the rates of reactions by the Arrhenius equation:

$$A_j = Q_j e^{\frac{-E_j}{RT}}, \tag{10}$$

where Q_j - the pre-exponential coefficient;

T - reaction temperature (in degrees Kelvin);

E_j - energy of activation (in kcal/mol);

R = 1.987·10^{-3} kcal/(deg·mol).

Methods for solving problems of physical-chemical hydrodynamics

Simulation of flow [2], the complicated physical and chemical processes is conveniently performed in the natural variables "velocity - pressure" as in this case, the vector of the velocity field are calculated in the algorithm for solving. To derive an equation for calculating the pressure field, it is necessary to equations (1) to apply the divergence operation, using equation (3). Let us write resulting equation:

$$\frac{\partial D}{\partial t} + \sum_{i=1}^{3} U_i \frac{\partial D}{\partial x_i} + \sum_{i=1}^{3} \left(\frac{\partial U_i}{\partial x_i}\right)^2 + 2\sum_{i \neq j} \frac{\partial U_i}{\partial x_i}\frac{\partial U_j}{\partial x_j} = \nu \Delta D - \frac{1}{\rho}\Delta \mathrm{P} + \mathrm{div}\vec{\mathrm{F}}, \tag{11}$$

where $\vec{\mathrm{F}} = \beta\theta\vec{g}\vec{n}$ and

$$D = div\,\vec{V} + \sum_{i=1}^{3} \frac{\partial U_i}{\partial x_i}. \tag{12}$$

Boundary conditions for solid and fixed boundary surfaces is the vanishing of velocity components (conditions of the first kind) and the normal component of the pressure gradient (the condition of the second kind):

$$\left.\frac{\partial P}{\partial x_i}\right|_{\partial\Omega} = 0 \tag{13}$$

In that case there are inlets and outlets, the pressure are used for the boundary conditions of the first kind, which are determined directly. For the velocity components being asked parabolic profile for the velocity projection on an axis parallel to the hole, and vanishes projection perpendicular to it.

To improve numerical stability are used in the approximation of the convective terms counter flow derivatives:

$$U_x \frac{\partial S}{\partial x} \Rightarrow \begin{cases} U_{x,i,j,k} \frac{S_{i,j,k} - S_{i-1,j,k}}{h_x}, & U_{x,i,j,k} \geq 0; \\ U_{x,i,j,k} \frac{S_{i+1,j,k} - S_{i,j,k}}{h_x}, & U_{x,i,j,k} < 0. \end{cases} \tag{14}$$

The resulting difference scheme for finding of the pressure field:

$$\frac{P_{i-1,j,k} - 2P_{i,j,k} + P_{i+1,j,k}}{h_x^2} + \frac{P_{i,j-1,k} - 2P_{i,j,k} + P_{i,j+1,k}}{h_y^2} + \frac{P_{i,j,k-1} - 2P_{i,j,k} + P_{i,j,k+1}}{h_z^2} =$$

$$= -\rho\left(-\frac{D_{i,j,k}^m}{\tau} + \frac{U_{x,i,j,k}^m + \left|U_{x,i,j,k}^m\right|}{2}\frac{D_{i,j,k}^{m+1} - D_{i-1,j,k}^{m+1}}{h_x} + \frac{U_{x,i,j,k}^m - \left|U_{x,i,j,k}^m\right|}{2}\frac{D_{i+1,j,k}^{m+1} - D_{i,j,k}^{m+1}}{h_x} + \right.$$

$$+\frac{U^m_{y,i,j,k}+\left|U^m_{y,i,j,k}\right|}{2}\frac{D^{m+1}_{i,j,k}-D^{m+1}_{i,j-1,k}}{h_y}+\frac{U^m_{y,i,j,k}-\left|U^m_{y,i,j,k}\right|}{2}\frac{D^{m+1}_{i,j+1,k}-D^{m+1}_{i,j,k}}{h_y}+$$

$$+\frac{U^m_{z,i,j,k}+\left|U^m_{z,i,j,k}\right|}{2}\frac{D^{m+1}_{i,j,k}-D^{m+1}_{i,j,k-1}}{h_z}+\frac{U^m_{z,i,j,k}-\left|U^m_{z,i,j,k}\right|}{2}\frac{D^{m+1}_{i,j,k+1}-D^{m+1}_{i,j,k}}{h_z}+$$

$$+\left(\frac{U^m_{i-1,j,k}-U^m_{i+1,j,k}}{2h_x}\right)^2+\left(\frac{U^m_{i,j-1,k}-U^m_{i,j+1,k}}{2h_y}\right)^2+\left(\frac{U^m_{i,j,k-1}-U^m_{i,j,k+1}}{2h_z}\right)^2+$$

$$+2\left(\frac{U^m_{i,j-1,k}-U^m_{i,j+1,k}}{2h_y}\right)\left(\frac{U^m_{i-1,j,k}-U^m_{i+1,j,k}}{2h_x}\right)+$$

$$+2\left(\frac{U^m_{i,j,k-1}-U^m_{i,j,k+1}}{2h_z}\right)\left(\frac{U^m_{i-1,j,k}-U^m_{i+1,j,k}}{2h_x}\right)+$$

$$+2\left(\frac{U^m_{i,j,k-1}-U^m_{i,j,k+1}}{2h_z}\right)\left(\frac{U^m_{i,j-1,k}-U^m_{i,j+1,k}}{2h_y}\right)-$$

$$-\left(\frac{F^m_{i-1,j,k}-F^m_{i+1,j,k}}{2h_x}+\frac{F^m_{i,j-1,k}-F^m_{i,j+1,k}}{2h_y}+\frac{F^m_{i,j,k-1}-F^m_{i,j,k+1}}{2h_z}\right)-$$

$$-\nu\left(\frac{D^{m+1}_{i-1,j,k}-2D^{m+1}_{i,j,k}+D^{m+1}_{i+1,j,k}}{h_x^2}+\frac{D^{m+1}_{i,j-1,k}-2D^{m+1}_{i,j,k}+D^{m+1}_{i,j+1,k}}{h_y^2}+\frac{D^{m+1}_{i,j,k-1}-2D^{m+1}_{i,j,k}+D^{m+1}_{i,j,k+1}}{h_z^2}\right)\Bigg).$$

Parallelization of the computational process

Typically, computer programs for complex computations are performed on multiprocessor computers [3-5]. This is due to the fact that the solution is sufficiently complex problems, such as modeling of three-dimensional flow, the complicated physical and chemical processes on a single machine is difficult. Impact of limited resources (memory), and limited computing power. Even modern 4 and 8-core processors can solve complex problems for a very long time. Therefore obvious is the use of multiple processors, solving at the same time different parts of the problem. This situation is some degree due to the fact that many computational problems, such as physics and other fields of science can be broken down into smaller elementary problems that are solved simultaneously.

There are the following types of parallelism:

• geometric when the whole computational domain is divided into sections between processors;

• functional when the problem is divided into blocks, each of which performed by a single processor in parallel to other blocks;

• conveyor, which is similar to the geometric and functional types, difference lies in the fact that the implementation of computing begins first processor, then it passes the data to the second processor, which begins its calculations and at this time the first proceeds to the next step etc.

The most common systems in the world of parallel programming are MPI and OpenMP, and in them by a completely different approach to parallel programming. The first system (MPI) - a system with a divided memory, and the second (OpenMP) - shared with.

The MPI standard is suitable for use on cluster systems, as it hides the networking features of the processes provided by the API. OpenMP not designed to provide networking, it is used on a multi-processor or multi-core computer, providing the ability to dynamically use the available capacity of a supercomputer.

In the field of scientific computing become much more popular calculations using graphics accelerators. Modern graphics accelerators are massively parallel processors with shared memory.

In contrast to the central processor with multiple cores, a graphics processor contains a few dozen to a thousand cores that can perform calculations in parallel.

The most advanced technology at the moment is a system CUDA proposed by NVidia in 2007. It allows you to simply use graphics accelerators as massively parallel processors (SIMD) with a simple and understandable execution model and capacity utilization graphics accelerator. In this technology, the calculations are performed a plurality of blocks consisting of a number of processing threads.

Unlike MPI, NVidia CUDA is a system with shared memory. At the same time each access to shared memory, resulting in significant delays during which the calculation of the flow is not provided.

To avoid such delays, each core has a certain amount of shared memory, which is a fast shared memory for all threads of one block. Currently, any graphics accelerator implements streaming computational model - is a stream of input and output data, composed of the same elements which can be handled independently. Processing elements are kernel.

A distinctive feature of CUDA is that it is not possible to implement global synchronization between the computing units in any other way than to complete the calculations in all the blocks and transfer data to your computer. In this regard, each step of the computational algorithm can use their own, optimal for a division of tasks between units.

However, this feature cannot be fully transferred iterative algorithms on graphics accelerators, since each block after iteration on your site data and calculate the error should correlate them with other units, and this is not possible. Thus, iterative algorithms for a natural solution is to calculate one iteration of the device, and check the convergence and acceptance of the decision on the extension of the calculation receives a CPU. Furthermore, this approach solves the problem of boundary values for each block because they are updated each time you start the iteration.

There are approaches that artificially on the GPU is performed several iterations without global convergence check and update the boundary data from other units. Instead, the boundary values do not change anything, either interpolated or calculated for each inner iteration. After the end of inner iterations, and the need to bring data synchronization between different blocks. In this case, there may be situations where different blocks converge at different speeds or not joined together.

Therefore require special methods of resolving these conflicts. Naturally, this approach reflects badly on the general accuracy of the solution.

Modeling of fluid flow, complicated by the set chemical reactions, a large number of substances, presupposes a large computational grids for more detailed study. Furthermore, the number of substances and reactions between them can reach tens of thousands. In this case, the use of graphics accelerators becomes justified, as it allows them to fully load calculations.

However, on very large grids or in the transition to modeling in three dimensions becomes apparent and necessary to use the system with multiple graphics cards or even a cluster of such systems.

The use of accelerated methods of integrating chemical equations allows for faster get the results of mathematical modeling, to try different options in less time, to investigate the behavior of the system at the critical values of the parameters.

References

1. Anderson, J. Computational fluid dynamics: the basics with applications I. New York: McGraw-Hill, Inc., 1995.

2. Ferziger, J.H**.** Computational Methods for Fluid Dynamics. M. Peric.-Berlin: Springer-Verlag, 2002.

3. Zhang, Y. Fast tridiagonal solvers on the GPU // Proceedings of the 15th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP). - 2010. - P. 127-136.

4. Cohen, J. A Fast Double Precision CFD Code using CUDA // Parallel Computational Fluid Dynamics: Recent Advances and Future Directions. - 2009. P. 414-430.

5. Jacobsen, D.A. An MPI-CUDA Implementation for Massively Parallel Incompressible Flow Computations on Multi-GPU Clusters // 48th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition (4-7 January 2010, Orlando, Florida). - paper no: AIAA 2010-522. - 2010.

МОДЕЛИРОВАНИЕ ФИЗИКО-ХИМИЧЕСКИХ ПРОЦЕССОВ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИИ CUDA

А. Кабдолдина, Е. Тулешов, Ж. Уалиев
Институт проблем горения

Аннотация
Применение численных методов становится все более и более популярным в связи с растущей трудоемкостью экспериментов и сложностью получения подробной информации о потоках и химических процессах, происходящих в них. Например: при изучении химических лазеров, моделирования процессов, происходящих в результате пожара. Данная работа описывает развитие системы быстрых алгоритмов параллельных CUDA для решения разностных уравнений (Навье - Стокса, Пуассона) и химической кинетики, получаемых при решении задач моделирования течений с протекающими в них реакциями множества веществ.

CUDA ТЕХНОЛОГИЯСЫ БОЙЫНША ФИЗИКА-ХИМИЯЛЫҚ ПРОЦЕССТЕРДІ МОДЕЛДЕУ

А.Кабдолдина, Е.Тулешев, Ж. Уалиев

Жану проблемалар институты, 050012, Қазақстан, Алматы, Бөгенбай батыр көш.,172

Аннотация

Химиялық процесстер кезіндегі қажетті ақпараттарды алуға байланысты эксперименттердің қиындығы мен алынған ақпараттарды өңдеудің қиындығына байлансты қазіргі таңда сандық әдісін қолдану танымал болуда. Мысалы: химиялық лазерді зерттеу, өрт кезіндегі процестерді моделдеу. Аталған жұмыс CUDA параллель жедел алгоритмде жүйені әр түрлі теңдеулерде (Навье- Стокс, Пуассон) және реакция кезіндегі түзілетін өнімдердің химиялық кинетикасы арқылы моделдеудің көптеген есептерін шығаруға мүмкіндіктер алуға болады.