COMPUTATIONAL MATHEMATICS

MSC 35G50 **DOI: 10.14529/jcem230201**

SOLVING OF THE BIOLOGICAL KINETICS PROBLEM ON A HETEROGENEOUS MULTIPROCESSOR COMPUTER SYSTEM

Y. V. Belova[1], yvbelova@yandex.ru,
A. M. Atayan[1], atayan24@mail.ru,
V. N. Litvinov[1], LitvinovVN@rambler.ru,
A. E. Chistyakov[1], cheese_05@mail.ru,
E. O. Rakhimbaeva[1], lena_rahimbaeva@mail.ru,
A. V. Nikitina[1], nikitina.vm@gmail.com
[1]Don State Technical University, Rostov-on-Don, Russian Federation

The process of solving the problem of biological kinetics, which reduces to the numerical solution of the three-dimensional diffusion-convection equation, is described. The solution of the grid equations obtained as a result of the continuous model discretization is performed on the basis of the adaptive alternating-triangular method. The problem under consideration was numerically implemented on a distributed memory computing system designed for massively parallel computing, which significantly reduced the operating time of the software module. The study of the parallel calculations model on the GPU for a sequence of thickening grids with a different number of threads allowed us to develop the optimal mode of operation of parallel algorithms with a different amount of input data. A description of the software implementation of a mathematical model adapted for hybrid computer systems is given.

Keywords: mathematical modeling; explicit-implicit difference scheme; approximation error; computational domain decomposition; distributed memory computing system.

Introduction

In the field of mathematical modeling of the movement's processes of pollutants in water systems, as well as in the development of numerical methods for solving the problems, a situation has developed in which the conducted research considers individual phenomena and does not cover them in a complex [1, 2]. Therefore, to solve problems, it is necessary to develop and theoretically study new algorithms and programs for solving model problems, including the equations of aero- and hydrodynamics, satisfying the basic laws of conservation of matter, considering the multicomponence the medium [3, 4, 5, 6]. The grid-characteristic method is proposed to solve this class of problems.

Due to the large amount of computational work on grids containing millions of computational nodes, considering the complex geometry of the computational domain, it becomes necessary to develop parallel algorithms for the numerical solution of mathematical physics problems. An important condition is the presence of parallelism in the problem, which is just characteristic of the hydrobiological problems being solved. One of the ways to parallelize computing is the use of MPI technology, as well as

MPI+OpenMP hybrid technology, which allows you to exchange messages between processes performing the same task. In the work [7], an improved preconditioner for solving SLAEs by the conjugate gradient method and a method for using combinations of MPI and OpenMP technologies for its construction and handling is proposed. The conclusion about the expediency of switching to hybrid parallelization combining MPI and OpenMP technologies is also made in the work [8], where various aspects of creating parallel software for domain decomposition methods are considered. It is shown that as the number of subdomains increases, the number of iterations of the method used increases: when moving from 16 subdomains to 128, the number of iterations increases from 138 to 183 and at the same time the number of vectors stored during the program increases. At the same time, 67% of the time is devoted to the work of the non—parallel part of the program, 14% of the time is spent on MPI work, and about 18% of the time is spent on OpenMP work. In the paper [9] describes the results of solving three-dimensional Navier–Stokes equations on hybrid computing systems built using a combination of MPI and CUDA technologies, which reduced the total calculation time by 12 times. Comparison of various combinations of parallelization technologies in paper [10] on the example of solving the non-stationary problem of liquid filtration to wells with a complex trajectory in a three-dimensional region showed that the most effective combination is MPI+boost+CUDA, which allows to achieve acceleration up to 24 times compared to MPI-implementation.

In this regard, parallel algorithms for the implementation of the biological kinetics problem have been developed, focused on heterogeneous computing systems with distributed memory.

In parallel implementation, methods of decomposition of grid regions were developed to solve computationally time-consuming diffusion-convection problems, considering the architecture and parameters of a heterogeneous multiprocessor computing system.

1. Formulation of the Problem

Consider an approximation of the three-dimensional diffusion-convection equation (1):

$$c'_t + uc'_x + vc'_y + wc'_z = (\mu c'_x)'_x + (\mu c'_y)'_y + (\nu c'_z)'_z + f, \tag{1}$$

where u, v, w are components of a velocity vector, μ is coefficient of diffusion (turbulent) exchange, f is a function describing the intensity and distribution of sources.

Let's build a uniform grid ω_τ in time increments τ:

$$\omega_\tau = \left\{ t_n = n\tau,\ n = \overline{0, N_t},\ \ N_t\tau = T \right\}.$$

To solve the problem (1) we can use a scheme with weights:

$$\frac{c^{n+1} - c^n}{\tau} + u\left(c^{n+\sigma}\right)'_x + v\left(c^{n+\sigma}\right)'_y + w\left(c^{n+\sigma}\right)'_z =$$
$$= \left(\mu\left(c^{n+\sigma}\right)'_x\right)'_x + \left(\mu\left(c^{n+\sigma}\right)'_y\right)'_y + \left(\nu\left(c^{n+\sigma}\right)'_z\right)'_z + f^{n+\sigma},$$

where $c^{n+\sigma} = \sigma^{n+1} + (1-\sigma)\, c^n$, σ is scheme weight.

The solution of grid equations is performed on the basis of the adaptive alternating triangular method [11, 12]. The advantage of this approach compared to the explicit scheme

is the use of large steps in the transition between time layers at a given accuracy. The disadvantage of this approach is the high complexity of the transition between layers (transitions are carried out iteratively).

If the steps along one of the spatial coordinates are significantly smaller than the steps along the others, for example, when solving problems of heat and mass transfer in shallow reservoirs, the dimensions of the calculated area in the vertical direction can be hundreds to thousands of times smaller than the horizontal dimensions. To solve the problem (1) based on difference schemes with relatively small labor costs for the transition between time layers, compared with the explicit scheme (1.5 - 2 times larger), with large time steps (about 30 times more), we will use splitting schemes for two-dimensional and one-dimensional problems [13]:

$$\frac{c^{n+1/2}-c^n}{\tau}+u\left(c^n\right)'_x+v\left(c^n\right)'_y=\left(\mu\left(c^n\right)'_x\right)'_x+\left(\mu\left(c^n\right)'_y\right)'_y, \tag{2}$$

$$\frac{c^{n+1}-c^{n+1/2}}{\tau}+w\left(c^{n+(\sigma+1)/2}\right)'_z=\left(\nu\left(c^{n+(\sigma+1)/2}\right)'_z\right)'_z+f^{n+(\sigma+1)/2}, \tag{3}$$

where $c^{n+(\sigma+1)/2}=\sigma^{n+1}+(1-\sigma)\,c^{n+1/2}$, σ – scheme weight.

For numerical implementation a spatial grid is introduced for a discrete mathematical model of the problem:

$$\overline{w}_h=\{x_i=ih_x,\ y_j=jh_y;\ \ i=0,...,N_x,\ j=0,...,N_y;\ N_t\tau=T,\ N_xh_x=l_x,\ N_yh_y=l_y\}$$

where $h_x,\ h_y$ are steps through the space, $N_x,\ N_y$ are boundaries by space, N_t is upper bound on time, τ is time step, $l_x,\ l_y$ are characteristic dimensions of the computational domain.

To approximate a homogeneous equation (2) we will use splitting schemes in spatial coordinate directions:

$$\frac{c^{n+1/4}-c^n}{\tau}+u(c^n)'_x=\left(\mu(c^n)'_x\right)'_x, \tag{4}$$

$$\frac{c^{n+1/2}-c^{n+1/4}}{\tau}+v\left(c^{n+1/4}\right)'_y=\left(\mu\left(c^{n+1/4}\right)'_y\right)'_y.$$

1.1. Difference Schemes for Solving Convection-Diffusion-Reaction Equations

To solve real problems of hydrophysics of shallow water bodies, three-layer difference schemes based on a linear combination of Upwind and Standard Leapfrog difference schemes with weighting coefficients 2/3 и 1/3, respectively, obtained as a result of minimizing the order of approximation error, are applied. These schemes, when solving the diffusion-convection problem, have a lower grid viscosity and, as a result, more accurately describe the behavior of the solution in the case of large grid numbers of Peclet (up to 20–50) [14]. The three-layer difference scheme used has greater accuracy than the traditional Standard Leapfrog scheme when solving problems in which convection prevails over diffusion.

To increase the accuracy of calculations, schemes that consider the fullness of the calculation cells will also be used [15]. The evaluation of the accuracy of the numerical solution of the hydrodynamics problem on a sequence of thickening computational grids in

the case of direct use of rectangular grids (stepwise approximation of boundaries) showed that the relative error of calculations can reach up to 70% of the exact solution; under the same conditions, the use of the proposed method reduces the error to 6%. Splitting a rectangular grid 2–8 times in each of the spatial directions does not lead to the same increase in accuracy that numerical solutions obtained considering the method of partial cell occupancy have.

To approximate the system of equations (4), we will use the scheme obtained as a result of a linear combination of the Upwind and Standard Leapfrog schemes, while considering the function of cell occupancy [12, 15]:

– the difference scheme for the equation (4), describing the transfer along the Ox, direction will be written as:

$$\frac{2q_{2,i,j}+q_{0,i,j}}{3}\frac{c_{i,j}^{n+1/4}-c_{i,j}^{n}}{\tau}+5u_{i-1/2,j}q_{2,i,j}\frac{c_{i,j}^{n}-c_{i-1,j}^{n}}{3h_x}+u_{i+1/2,j}\min\left(q_{1,i,j},q_{2,i,j}\right)\frac{c_{i+1,j}^{n}-c_{i,j}^{n}}{3h_x}+$$

$$+\frac{2\Delta_x c_{i-1,j}^{n}q_{2,i,j}+\Delta_x c_{i,j}^{n}q_{0,i,j}}{3}=2\mu_{i+1/2,j}q_{1,i,j}\frac{c_{i+1,j}^{n}-c_{i,j}^{n}}{h_x^2}-2\mu_{i-1/2,j}q_{2,i,j}\frac{c_{i,j}^{n}-c_{i-1,j}^{n}}{h_x^2}-$$

$$-\left|q_{1,i,j}-q_{2,i,j}\right|\mu_{i,\,j}\frac{\alpha_x c_{i,j}^{n}+\beta_x}{h_x},\ u_{i,j}\geq 0,\text{where }\Delta_x c_{i,j}^{n}=\frac{c_{i,j}^{n-3/4}-c_{i,j}^{n-1}}{\tau}.$$

– the difference scheme for the equation (4), describing the transfer along the direction Oy, direction will be written as:

$$\frac{2q_{4,i,j}+q_{0,i,j}}{3}\frac{c_{i,j}^{n+1/2}-c_{i,j}^{n+1/4}}{\tau}+5v_{i,j-1/2}q_{4,i,j}\frac{c_{i,j}^{n+1/4}-c_{i,j-1}^{n+1/4}}{3h_y}+$$

$$+v_{i,j+1/2}\min\left(q_{3,i,j},q_{4,i,j}\right)\frac{c_{i,j+1}^{n+1/4}-c_{i,j}^{n+1/4}}{3h_y}+\frac{2\Delta_y c_{i,j-1}^{n+1/4}q_{4,i,j}+\Delta_y c_{i,j}^{n+1/4}q_{0,i,j}}{3}=$$

$$=2\mu_{i,j+1/2}q_{3,i,j}\frac{c_{i,j+1}^{n+1/4}-c_{i,j}^{n+1/4}}{h_y^2}-2\mu_{i,j-1/2}q_{4,i,j}\frac{c_{i,j}^{n+1/4}-c_{i,j-1}^{n+1/4}}{h_y^2}-$$

$$-\left|q_{3,i,j}-q_{4,i,j}\right|\mu_{i,\,j}\frac{\alpha_y c_{i,j}^{n+1/4}+\beta_y}{h_y}v_{i,j}\geq 0,\text{where }\Delta_y c_{i,j}^{n+1/4}=\frac{c_{i,j}^{n-1/2}-c_{i,j}^{n-3/4}}{\tau}.$$

Here $q_0,\ q_1,\ q_2,\ q_3,\ q_4$ are coefficients describing the degree of occupancy of the control areas.

In order to obtain difference schemes approximating the system of equations (4) for $u_{i,j}<0$ and $v_{i,j}<0$, the corresponding coordinate axes Ox and Oy are necessary from the approximations presented send it in opposite directions. The equation (3) is solved by the run-through method [11].

For solving non-stationary problems, the splitting scheme into two-dimensional and one-dimensional problems has an advantage. In this case, the two-dimensional problem is solved on the basis of explicit schemes, and the one-dimensional one is approximated by schemes with weights and solved by the run-through method. When solving stationary problems, schemes with weights are used. Using this approach allows us to reduce the

original problem to solving grid equations, for which iterative methods are used. In the class of two-layer iterative methods, one of the most effective is the adaptive alternating-triangular method [11].

2. Parallel Implementation of the Method of the Biological Kinetics Problem Solving

2.1. Decomposition of the Computational Domain in one Spatial Direction

We describe the construction of parallel algorithms for solving the problem (4) using the scheme described in section 1.1 in horizontal directions. The values of the field are calculated in the nodes of the calculated grid $c(x, y)$: $c_{i,j}$ by $i \in \overline{1, N_x - 2}$, $j \in \overline{1, N_y - 2}$, at the same time, along the perimeter ($i \in \{0, N_x - 1\}$, $j \in \{0, N_y - 1\}$) there are fictitious nodes. Let us decompose the computational domain along the spatial direction Oy with lines, which are parallel to the spatial direction Ox, meanwhile, let us denote w^r is subdomain with number r , $0 \leqslant r \leqslant p-1$, where p is the number of subareas into which the original area is divided. Settlement nodes of the region w^r are the elements $c^r_{i,j}$ at $i \in \overline{1, N_x - 2}$, $j \in \overline{1, N_2^r - 2}$. The splitting of the original area is done in such a way that adjacent areas w^r and w^{r+1} intersect at two nodes along the direction perpendicular to the partition lines, and the equalities $c^r_{i,N_2^r-2} = c^{r+1}_{i,0}$, $c^r_{i,N_2^r-1} = c^{r+1}_{i,1}$ (Fig. 3). To represent a field value $c(x, y)$ in vector form a pair of indices $(i,\ j)$ value can be matched m, describing the ordinal number of the elements of the vector u: $m = i+jN_x$, $0 \leqslant m \leqslant n-1$, n – vector length $c = (c_0, c_1, ..., c_{n-1})^T$. This representation is convenient to use when describing and researching algorithms for solving grid equations by iterative methods.

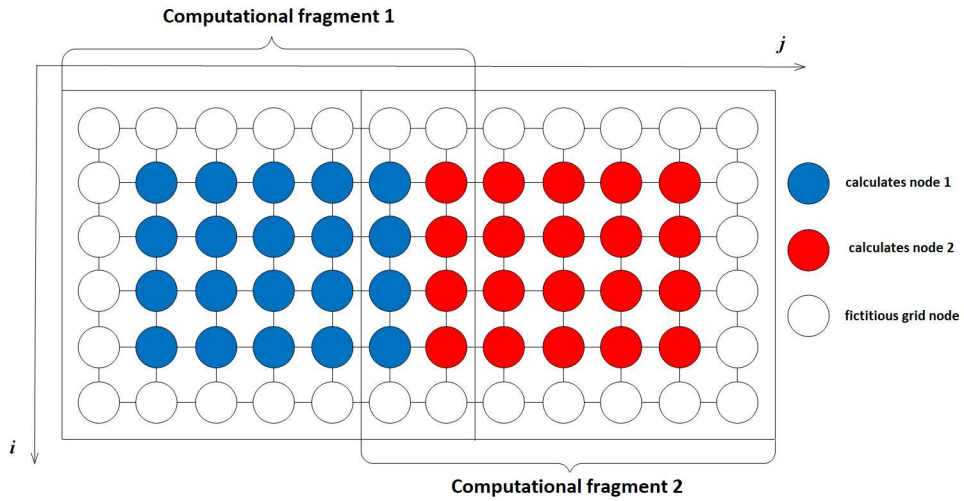

Fig. 1. Decomposition of the computational domain

For fragments w^r, obtained as a result of decomposition of the computational domain in one spatial direction, it is necessary to know two parameters: the initial index $j = N_1^r$ in the initial computational domain and the width of the fragment N_2^r. The index number N_1^r from which the corresponding fragment of the computational domain begins, can be calculated using the formula

$$N_1^r = \lfloor r \cdot (N_y - 2)/p \rfloor ,$$

where $\lfloor x \rfloor$ – the function «floor» is defined as the largest integer less than or equal to x, $\lceil x \rceil$ – the function «ceiling» is defined as the smallest integer greater than or equal to x. The width of the subdomain w^r along the axis Oy is calculated by the formula

$$N_2^r = \lfloor (r+1) \cdot (N_y - 2)/p \rfloor - N_1^r + 2.$$

The following parameters are used for the theoretical evaluation of the work of computing systems:

– t_a – execution time of one arithmetic operation;

– t_l – time of data transmission organization (latency);

– t_x – transmission time of one given.

Figure 4 shows a graph of the dependence of the transmission time on the amount of data for a different number of exchanges between nodes of the computing system. The graph shows that the transmission time dependence function has a jump when the amount of transmitted data is approximately 512 floating point numbers. Let's denote this value $N_{\max} = 512$.

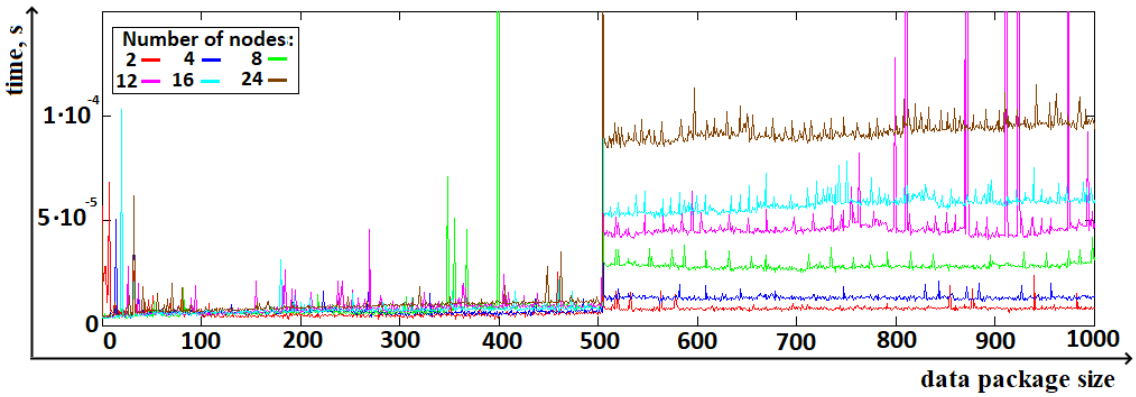

Fig. 2. Dependence of data transfer time on volume when working with a different number of computing nodes

The calculation of data on the computer system used allows to significantly reduce the calculation time. However, the efficiency of the computing system's operating time may not always be expected. In this case, it is correct to carry out a theoretical analysis of calculating the calculation time based on regression analysis. Consider the multiple regression model. Vector t_l is the total operating time of the computing system (seconds), vectors n, p are explanatory factors: the amount of data transmitted and the number of computing nodes used. For the latency time, the formula holds:

$$t_l(p,n) = \begin{cases} 5.21 \times 10^{-6} + 1.53 \times 10^{-7} p, & \text{if } n \leqslant 512; \\ 6.733 \times 10^{-6} p, & \text{if } n > 512. \end{cases} \tag{5}$$

Transmission time of one given $t_x = 3.3 \times 10^{-9}$.

Calculating the transfer along horizontal directions based on explicit difference schemes using a six–point template requires $10N$ arithmetic operations, where N is the number of nodes of the computational grid. The calculation of the transfer along the vertical direction requires $13N$ arithmetic operations, of which $5N$ refers to the calculation of the right parts,

and $8N$ is required to solve the grid equation by the run-through method. Thus, the time spent on one iteration in the case of a sequential version of the algorithm will be written as:

$$t = 33t_a\left(N_x - 2\right)\left(N_y - 2\right). \tag{6}$$

Let's estimate the calculation time using a parallel algorithm on a distributed memory system:

$$t = 33t_a\left(N_x - 2\right)\max_r\left(N_2^r - 2\right) + 2\left(t_l\left(p, \left(N_x - 2\right)\right) + \left(N_x - 2\right)t_x\right),$$

$$\left\lfloor\frac{N_y - 2}{p}\right\rfloor \leqslant \max_r\left(N_2^r - 2\right) \leqslant \left\lceil\frac{N_y - 2}{p}\right\rceil, \max_r\left(N_2^r - 2\right) \approx \frac{N_y - 2}{p}.$$

If the amount of transmitted data is greater than $N_x - 2 > N_{\max}$, then $k = \left\lceil\frac{N_2-2}{N_{\max}}\right\rceil$ exchanges, then the time spent by the parallel algorithm is equal to:

$$t = 33t_a\frac{\left(N_x - 2\right)\left(N_y - 2\right)}{p} + 2\left(t_l\left(p, \lceil\left(N_x - 2\right)/k\rceil\right)k + \left(N_x - 2\right)t_x\right). \tag{7}$$

Acceleration of parallel operation of the algorithm is equal to:

$$A = p\Bigg/\left(1 + \frac{2p\left(t_l\left(p, \lceil\left(N_x - 2\right)/k\rceil\right)k + \left(N_x - 2\right)t_x\right)}{33t_a\left(N_x - 2\right)\left(N_y - 2\right)}\right). \tag{8}$$

To solve three-dimensional diffusion-convection problems in areas whose linear dimensions along one direction are significantly smaller than the dimensions in the remaining two spatial directions, which is typical, for example, for shallow reservoirs, it is necessary to decompose the calculated area in two spatial directions. When using decomposition in one direction, the volume of transfers is equal to $2 \cdot p \cdot Ny \cdot Nz$, where p is the number of computing nodes involved. In the case of decomposition in two directions, the volume of transfers is equal to $2 \cdot (p_x \cdot Ny + p_y \cdot Nx) \cdot Nz$, where Nx, Ny, Nz is the number of settlement nodes along the directions of the axes Ox, Oy and Oz, accordingly. The method of splitting into rectangles of p_x blocks along one direction and p_y blocks along the other is used.

A comparison of the algorithms is given for a different number of computational nodes with a variable decomposition of the computational domain. Table 1 shows the results of a parallel version of the algorithm based on MPI technology for the splitting scheme.

The effectiveness of parallel programs on distributed memory systems depends significantly on the communication environment. The communication environment is sufficiently fully characterized by two parameters: bandwidth, which determines the number of bytes transmitted per unit of time, and latency. Communication operations are performed much slower than accessing local memory, so those parallel programs in which exchanges are minimized will be the most effective.

2.2. Architecture of the Software Package

The software implementation of the mathematical model (1) is a console application written in C++, with support for CUDA technology. As a result of object-oriented analysis and design, a class library has been created that allows you to work with the description

of the geometry of the computational domain as a set of geometric primitives and generate computational grids based on it with the required decomposition parameters and arrays of coefficients of grid equations. The data required for the simulation is stored in a text file, which is then used by the solver. The results of the numerical solution are saved in a report file. The library is based on classes that allow you to read and write information about the geometry of the object under study and the parameters of the calculated grid. These classes must implement the IGeometrySerializer and IgridSerializer interfaces, respectively. Initialized objects of these types are used to create Geometry Geometry objects and a Grid calculation grid, on the basis of which an object of the GridGeometryPreCalculated class is created, which is a geometry object with a superimposed calculation grid and pre-calculated parameters, such as, for example, node characteristics: internal, boundary or dummy.

Table 1

Results of the parallel version of the algorithm based on MPI technology for the splitting scheme into explicit and implicit problems

p	p_x	p_y	Time, s	Boost	Effectiveness, %
1	1	1	38.16	1.00	100
2	2	1	19.6903	1.94	97
3	3	1	13.5332	2.82	94
4	2	2	10.001	3.82	95.5
	4	1	10.2104	3.74	93.5
8	4	2	5.2185	7.31	91.38
	8	1	5.36947	7.11	88.88
16	4	4	2.98691	12.78	79.88
	8	2	3.00383	12.70	79.38
	16	1	3.16188	12.07	75.44
20	5	4	2.499	15.27	76.35
	10	2	2.5475	14.98	74.90
	20	1	2.68232	14.23	71.15
24	6	4	2.24099	17.03	70.96
	8	3	2.24239	17.02	70.92
	12	2	2.24567	16.99	70.79
	24	1	2.50914	15.21	63.38

In order to provide the possibility of intermediate data storage for continuing calculations in case of emergency situations or transferring the task to other computing nodes, the GridGeometryPreCalculated object is connected by an aggregation relationship with the IgridGeometryPreCalculatedSerializer interface. Classes implementing the IgridGeometryPreCalculatedSerializer interface solve the tasks of serializing data in various formats (xml, json, storing data in relational databases, etc.)

The configured GridGeometryPreCalculated object is used when creating a SolverTask calculation task object, during which the user sets a number of modeling characteristics, the most important of which are the model parameters and the derivative calculation template.

Directly modeling is performed by the Solver class, which uses objects of the SolverTask class as tasks. The features of the Solver class are the possibility of dynamic control of computing devices by means of an aggregation relationship with an object of the IDeviceManager interface and the provision of work with an object of a system of linear algebraic equations by means of a composition relationship with an object of the SLAE class.

A distinctive feature of the SLAE class is working with the matrix object Matrix. The implementation of this interface in the work is the SparseMatrixRepSeq class, which provides efficient data storage with the elimination of duplication. The class diagram is shown in Figure 3.

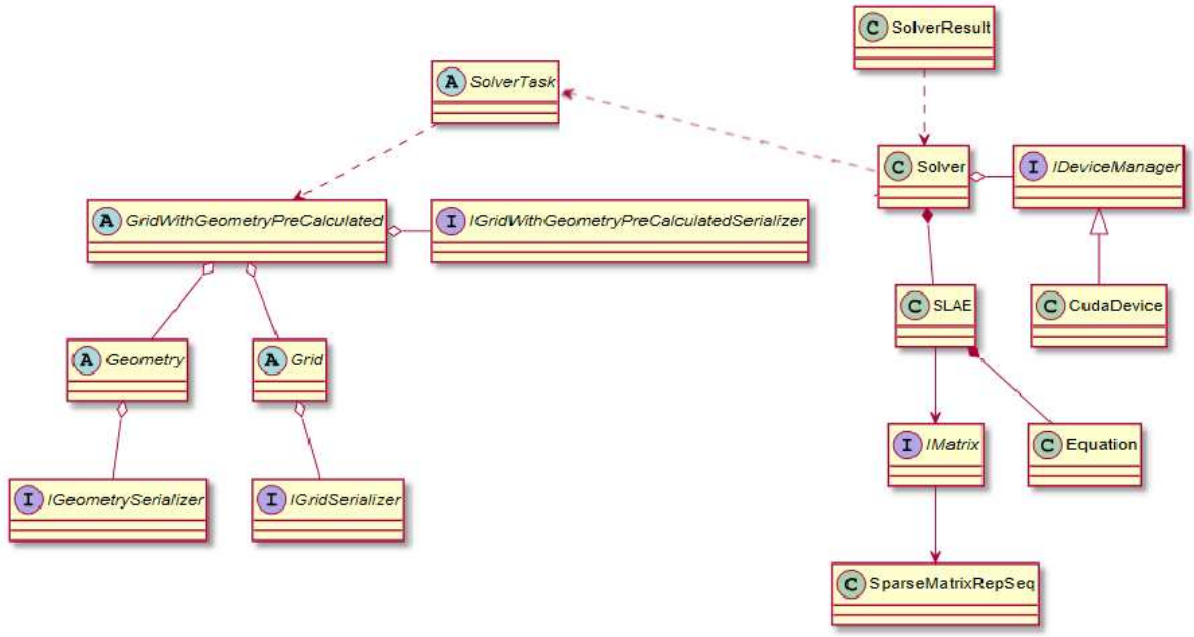

Fig. 3. Class diagram of the developed software package

2.3. Parallel Implementation of the Biological Kinetics Problem

The proposed mathematical model of biological kinetics is numerically implemented based on the development of parallel algorithms adapted for hybrid computer systems using the NVIDIA CUDA architecture [16]. Table 2 shows the results of a numerical solution of a two-dimensional biological kinetics problem for a different number of threads involved in the implementation of a parallel algorithm on CUDA, for experiments 1–7, p is the number of threads involved, t is the calculation time in seconds. Calculations were carried out on a sequence of rectangular grids: from 100×100 to 20000×20000 settlement nodes.

The developed parallel algorithms for solving the problem of biological kinetics on a sequence of thickening rectangular computational grids made it possible to increase the efficiency of the method for solving the problem. Numerical experiments made it possible to compare the efficiency of the proposed GPU-oriented algorithms in the case of different numbers of threads involved. For Experiment 6, the number of calculated nodes was 10^8, the minimum time on 128 threads was 23.9 s. Analysis of data from experiment 7 on a

Table 2

Results of the algorithm on CUDA

N	1	2	3	4	5	6	7
$Nx = Ny$	100	500	1000	2000	5000	10000	20000
t, c; p=128	0.004296	0.0704	0.26099	0.95916	6.4114	23.892	188.767
t, c; p=256	0.004252	0.0705	0.25349	0.97414	6.4082	24.393	220.024
t, c; p=512	0.004301	0.0716	0.25046	0.95707	6.5460	24.981	209.204
t, c; p=1024	0.004080	0.0745	0.25876	0.97980	6.4035	24.993	204.419

rectangular grid of 4×10^8 calculated nodes showed that the maximum operating time of the algorithm was 220 s with 256 threads involved. The results of numerical experiments on the NVIDIA Tesla K80 graphics accelerator have shown the advantage of running a parallel algorithm with a large number of computational nodes.

3. Conclusion

The development of effective parallel algorithms for the numerical implementation of the biological kinetics made it possible to study both intra- and interspecific chemical communications between planktonic populations of the coastal system – the Azov Sea in a limited time mode, which is relevant in the event of catastrophic environmental situations, which include eutrophication and exogenous hypoxia processes. It is established that the significant heterogeneity of the detritus structure caused by various fractions of natural organic substances plays an important role in regulating global processes of aquatic ecology. Parallel algorithms for solving the problem of biological kinetics have been developed, focused on a computing system with distributed memory, on a sequence of thickening grids. The study of the dependence of the transmission time on the amount of data for a different number of exchanges between the nodes of the computing system showed a jump in the amount of transmitted data equal to approximately 512 floating point numbers. Their effectiveness in the case of different number of involved flows is analyzed. The results of numerical experiments on the NVIDIA Tesla K80 graphics accelerator have shown the advantage of running a parallel algorithm with a large number of computational nodes.

The study was supported by the Russian Science Foundation grant No. 22-71-10102, https://rscf.ru/project/22-71-10102/.

References

1. Fennel K. The Generation of Phytoplankton Patchiness by Mesoscale Current Patterns. *Ocean Dynamics*, 2001, vol. 52, pp. 58–70.

2. Tyutyunov Yu.V., Zagrebneva A.D., Azovsky A.I. Spatiotemporal Pattern Formation in a Prey-Predator System: The Case Study of Short-Term Interactions Between Diatom Microalgae and Microcrustaceans. *Mathematics*, 2020, vol. 8, no. 7, article ID: 1065. DOI: 10.3390/math8071065.

3. Chetverushkin B.N. Limits of Detail and Formulation of Continuum Equation Models. *Matematicheskoe modelirovanie*, 2012, vol. 24, no. 1, pp. 33. (in Russian)
4. Yakushev E.V., Podymov O.I., Chasovnikov V.K. Seasonal Changes in the Hydrochemical Structure of the Black Sea Redox Zone. *Oceanography*, 2005, vol. 18, no. 2, pp. 48–55.
5. Weiner E.R. *Application of Environmental Chemistry: A Practical Guide for Environmental Professionals.* Boca Raton, London, New York, Washington, D.C., CRC Press LLC, 2000, 281 p.
6. Treguer P., Legendre L., Rivkin R.T., Ragueneau O., Dittert N. Water Column Biogeochemistry Below the Euphotic Zone. *Ocean Biogeochemistry: The Role of the Ocean Carbon Cycle in Global Change.* Berlin, Springer-Verlag, 2003, pp. 145–156.
7. Milyukova O.Yu. MPI+OpenMP Implementation of Conjugate Gradients Method With Preconditioner of the Block Incomplete Inverse Triangular Decomposition of the First Order. *Numerical Methods and Programming*, 2022, vol. 23, no 3, pp. 191–206. DOI: 10.26089/NumMet.v23r312 (in Russian)
8. Gurieva Y.L., Perevozkin D.V. Issues of Parallel Software Development for the Domain Decomposition Methods. *Numerical Methods and Programming*, 2018, vol. 19, no 3, pp. 219–229. DOI: 10.26089/NumMet.v19r321 (in Russian)
9. Kudrjashov I.Ju., Luckij A.E. Adaptation of the Code for Calculating Viscous Fluid Flows for Hybrid Computing Systems Based on CUDA-MPI Technologies. *Matematicheskoe modelirovanie*, 2012, vol. 24, no 7, pp. 33–44.
10. Gubajdullin D.A., Nikiforov A.I., Sadovnikov R.V. On the Peculiarities of Using the Architecture of a Heterogeneous Cluster for Solving Problems of Continuum Mechanics. *Numerical Methods and Programming*, 2011, vol. 12, no. 4, pp. 450–460.
11. Samarskij A.A., Vabishhevich P.N. *Chislennye metody reshenija zadach konvekcii-diffuzii [Numerical Methods for Solving Convection-Diffusion Problems].* M., URSS, 2009. – 246 p. (in Russian)
12. Sukhinov A.I., Chistyakov A.E., Alekseenko E.V. Numerical Realization of Three-Dimensional Model of Hydrodynamics for Shallow Water Basins on High-Performance System. *Mathematical Models and Computer Simulations*, 2011, vol. 3, pp. 562–574. DOI: 10.1134/S2070048211050115
13. Goloviznin, V.M., Samarskij A.A. Difference Approximation of Convective Transport With Spatial Splitting of the Time Derivative. *Matematicheskoe modelirovanie*, 1998, vol. 10, no. 1, pp. 86–100.
14. Suhinov A.I., Chistjakov A.E., Sidorjakina V.V., Procenko S.V. Economical Explicit-Implicit Schemes for Solving Multidimensional Diffusion-Convection Problems. *Computational Continuum Mechanics*, 2019, vol. 12, no. 4, pp. 435–445. (in Russian) DOI: 10.7242/1999-6691/2019.12.4.37
15. Sukhinov A., Chistyakov A., Kuznetsova I., Belova Y., Rahimbaeva E. Development and Research of a Modified Upwind Leapfrog Scheme for Solving Transport Problems. *Mathematics*, 2022, vol. 10, article ID: 3564.
16. Boreskov A.V., Harlamov A.A. *Basics with CUDA Technology* . Moscow, DMK Press, 2019. – 232 p. (in Russian)

Yulia V. Belova, PhD (Math), Department of Mathematics and Computer Science, Don State Technical University (Rostov-on-Don, Russian Federation), yvbelova@yandex.ru

Asya M. Atayan, Department of Computer software and automated systems, Don State Technical University (Rostov-on-Don, Russian Federation), atayan24@mail.ru

Vladimir N. Litvinov, PhD (Techn), Department of Mathematics and Computer Science, Don State Technical University (Rostov-on-Don, Russian Federation), LitvinovVN@rambler.ru

Alexander E. Chistyakov, DSc (Math), Department of Computer software and automated systems, Don State Technical University (Rostov-on-Don, Russian Federation), cheese_05@mail.ru

Elena O. Rakhimbaeva, Department of Computer software and automated systems, Don State Technical University (Rostov-on-Don, Russian Federation), lena_rahimbaeva@mail.ru

Alla V. Nikitina, Department of Computer software and automated systems, Don State Technical University (Rostov-on-Don, Russian Federation), nikitina.vm@gmail.com

Received April 7, 2023

УДК 517.9 **DOI: 10.14529/jcem230201**

РЕШЕНИЕ ЗАДАЧИ БИОЛОГИЧЕСКОЙ КИНЕТИКИ НА ГЕТЕРОГЕННОЙ МНОГОПРОЦЕССОРНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

Ю. В. Белова, А. М. Атаян, В. Н. Литвинов, А. Е. Чистяков, Е. О. Рахимбаева, А. В. Никитина

Описан процесс решения задачи биологической кинетики, сводящейся к численному решению трехмерного уравнения диффузии-конвекции. Решение сеточных уравнений, полученных в результате дискретизации непрерывной модели, выполняется на основе адаптивного попеременно-треугольного метода. Рассматриваемая задача была численно реализована на вычислительной системе с распределенной памятью, рассчитанной на массивно параллельные вычисления, что позволило значительно сократить время работы программного модуля. Проведенное исследование модели параллельных расчетов на GPU для последовательности сгущающихся сеток при разном количестве потоков позволило разработать оптимальный режим работы параллельных алгоритмов при разном объеме входных данных. Приведено описание программной реализации математической модели, адаптированной для гибридных компьютерных систем.

Ключевые слова: математическое моделирование; явно-неявная разностная схема; погрешность аппроксимации; декомпозиция расчетной области; вычислительная система с распределенной памятью.

Литература

1. Fennel, K. The Generation of Phytoplankton Patchiness by Mesoscale Current Patterns / K. Fennel // Ocean Dynamics. – 2001. – V. 52. – P. 58–70.
2. Tyutyunov, Yu.V. Spatiotemporal Pattern Formation in a Prey-Predator System: The Case Study of Short-Term Interactions Between Diatom Microalgae and Microcrustaceans / Yu.V. Tyutyunov, A.D. Zagrebneva, A.I. Azovsky // Mathematics. – 2020. – V. 8, № 7. – article ID: 1065.
3. Четверушкин, Б.Н. Пределы детализации и формулировка моделей уравнения сплошных сред / Б.Н. Четверушкин // Математическое моделирование. – 2012. – Т. 24, №1. – С. 33-52.
4. Yakushev, E.V. Seasonal Changes in the Hydrochemical Structure of the Black Sea Redox Zone / E.V. Yakushev, O.I. Podymov, V.K. Chasovnikov // Oceanography. – 2005. – V. 18, № 2. – P. 48–55.
5. Weiner, E.R. Application of Environmental Chemistry: A Practical Guide for Environmental Professionals / E.R. Weiner. – Boca Raton, London, New York, Washington, D.C.: CRC Press LLC, 2000. – 281 p.
6. Treguer, P. Water Column Biogeochemistry Below the Euphotic Zone / P. Treguer, L. Legendre, R. Rivkin, O. Ragueneau, N. Dittert // Ocean Biogeochemistry: The Role of the Ocean Carbon Cycle in Global Change. Berlin: Springer-Verlag, 2003. p. 145–156.
7. Милюкова, О.Ю. MPI+OpenMP реализация метода сопряженных градиентов с предобусловливателем блочного неполного обратного треугольного разложения первого порядка / О.Ю. Милюкова // Вычислительные методы и программирование. – 2022. – Т. 23, № 3. – С. 191–206.
8. Гурьева, Я.Л. Вопросы разработки параллельного программного обеспечения для метода декомпозиции области / Я.Л. Гурьева, Д.В. Перевозкин // Вычислительные методы и программирование. – 2018. – Т. 19, № 3. – С. 219–229.
9. Кудряшов, И.Ю. Адаптация кода для расчета течений вязких жидкостей под гибридные вычислительные системы на базе технологий CUDA-MPI / И.Ю. Кудряшов, А.Е. Луцкий // Математическое моделирование. – 2012. – Т. 24, № 7. – С. 33–44.
10. Губайдуллин, Д. А. Об особенностях использования архитектуры гетерогенного кластера для решения задач механики сплошных сред / Д.А. Губайдуллин, А.И. Никифоров, Р.В. Садовников // Вычислительные методы и программирование. – 2011. – Т. 12, № 4. – С. 450–460.
11. Самарский, А.А. Численные методы решения задач конвекции-диффузии / А.А. Самарский, П.Н. Вабищевич. – М.: URSS, 2009. – 246 с.
12. Сухинов, А.И. Численная реализация трехмерной модели гидродинамики для мелководных водоемов на супервычислительной системе / А.И. Сухинов, А.Е. Чистяков, Е.В. Алексеенко // Математическое моделирование. – 2011. – Т. 23, № 3. – С. 3–21.

13. Головизнин, В.М., Самарский А.А. Разностная аппроксимация конвективного переноса с пространственным расщеплением временной производной / В.М. Головизнин, А.А. Самарский // Математическое моделирование. – 1998. – Т. 10, № 1. – С. 86–100.
14. Сухинов, А.И. Экономичные явно-неявные схемы решения многомерных задач диффузии-конвекции / А.И. Сухинов, А.Е. Чистяков, В.В. Сидорякина, С.В. Проценко // Вычислительная механика сплошных сред. – 2019. – Т. 12, №4. – С. 435–445.
15. Sukhinov, A. Development and Research of a Modified Upwind Leapfrog Scheme for Solving Transport Problems / A. Sukhinov, A. Chistyakov, I. Kuznetsova, Y. Belova, E. Rahimbaeva // Mathematics. – 2022. – V. 10, article ID: 3564.
16. Боресков, А.В. Основы работы с технологией CUDA / А.В. Боресков, А.А. Харламов. – М.: ДМК Пресс, 2019. – 232 с.

Белова Юлия Валериевна, кандидат физико-математических наук, кафедра «Математика и информатика», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), yvbelova@yandex.ru

Атаян Ася Михайловна, кафедра «Программное обеспечение вычислительной техники и автоматизированных систем», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), atayan24@mail.ru

Литвинов Владимир Николаевич, кандидат технических наук, кафедра «Математика и информатика», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), LitvinovVN@rambler.ru

Чистяков Александр Евгеньевич, доктор физико-математических наук, кафедра «Программное обеспечение вычислительной техники и автоматизированных систем», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), cheese_05@mail.ru

Рахимбаева Елена Олеговна, кафедра «Программное обеспечение вычислительной техники и автоматизированных систем», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), lena_rahimbaeva@mail.ru

Никитина Алла Валерьевна, кафедра «Программное обеспечение вычислительной техники и автоматизированных систем», Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация), nikitina.vm@gmail.com

Поступила в редакцию 7 апреля 2023 г.