A software package for modeling the propagation of dynamic wave disturbances in heterogeneous multi-scale media

N. Khokhlov
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Russia
E-mail: kl@lokip.ru

Abstract. The paper considers a software package designed to simulate the propagation of dynamic wave disturbances in heterogeneous media. One of the main features of the considered software package is numerical algorithms with an explicit selection of inhomogeneities. Within the framework of the work, such inhomogeneities as pores, fractures and interfaces between media (contact boundaries) are considered. The considered algorithms make it possible to perform calculations in different scale settings in micro and macro sizes. The mathematical model is based on the equations of the linear theory of elasticity. For the calculation, block structural meshes are used. The software package is parallelized using MPI and OpenMP technologies. Separate parts of the algorithm are parallelized using graphics accelerators such as GPGPU. The paper describes the features of the algorithms under consideration and provides examples of calculations that demonstrate the capabilities of the algorithm.

1. Introduction
Numerical modeling of the propagation of dynamic wave disturbances in getherogenous medium is used to solve a wide range of problems. These tasks include, among other things, the tasks of seismic [1], strength of materials, high-speed loads [2], and tasks of non-destructive testing. The role of numerical modeling in each of these areas is very important. Thus, the numerical modeling of the propagation of seismic waves is an essential part of the work in geological exploration in the oil industry. Mathematical modeling is carried out in various geological environments [3], including in layered environments and in environments with the presence of various inhomogeneities (for example, fractures or caverns) [4][5]. Tasks of this kind seem to be very resource-intensive in terms of computational resources. The computation area is usually a seismic cube with an edge length of 1 km to 10 km. At the same time, irregularities can be several meters in size. When modeling seismic resistance problems, one also has to deal with many inhomogeneities. The parameters of buildings, such as the thickness of the walls, the dimensions of the openings, are much smaller than the dimensions of the computational regions, which sometimes include large rock masses, more than 10 km in size along one direction. Thus, the computational mesh must be detailed enough to be able to correctly select all inhomogeneities. To obtain a sufficient calculation accuracy and take into account a large number of inhomogeneities, the use of large computational grids is required; in real calculations, meshes with sizes up to several tens of billions of nodes are used. Another important task is the problem of non-destructive testing and ultrasonic diagnostics. To isolate
all inhomogeneities and describe the complex structure of the considered media, the use of appropriate algorithms and methods is required [6].

2. Mathematical model
As mentioned earlier, the approaches under consideration are based on the equation of the linear theory of elasticity. According to [7], the state of a continuous linear elastic medium obeys the following equations:

\[ \rho \dot{v} = (\Delta \cdot T)^T, \]  

(1)

\[ T = \lambda (\Delta \cdot v) I + \mu (\Delta \otimes v + (\Delta \otimes v)^T), \]  

(2)

where \( \rho \) is density, \( v \) is velocity, \( T \) is the stress tensor, and \( \lambda, \mu \) are the Lame’s parameters, characterizing the elastic properties of the medium.

The grid-characteristic method uses the characteristic properties of the systems of hyperbolic equations, describing the elastic wave propagation [8, 9]. The mathematical principles of the GCM based on representing the equations of motion of the linear-elastic medium in the following form:

\[ q_t + A_1 q_x + A_2 q_y + A_3 q_z = 0. \]  

(3)

In the last equation, \( q \) is a vector of unknown fields, having 9 components and equal to

\[ q = [v_1 \ v_2 \ v_3 \ T_{11} \ T_{22} \ T_{33} \ T_{23} \ T_{13} \ T_{12}]^T. \]  

(4)

Matrices \( A_k, k=1,2,3 \), are the square 9 \( \times \) 9 matrices. Matrix \( A_1 \) is given by the following expression:

\[ A_1 = \begin{bmatrix}
0 & 0 & 0 & \rho^{-1} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \rho^{-1} \\
0 & 0 & 0 & 0 & 0 & 0 & \rho^{-1} & 0 & 0 \\
\lambda + 2\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \mu & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}. \]  

(5)

Matrix \( A_2 \) is given by the following expression:

\[ A_2 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & \rho^{-1} & 0 \\
0 & 0 & 0 & \rho^{-1} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \rho^{-1} & 0 & 0 & 0 & 0 \\
0 & \lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda + 2\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \mu & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \mu & 0 & 0 & 0 & 0 & 0 & 0 \\
\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}. \]  

(6)
and matrix $A_3$ can be written as follows:

$$A_3 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & \rho^{-1} & 0 \\
0 & 0 & 0 & 0 & 0 & \rho^{-1} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \rho^{-1} & 0 & 0 & 0 & 0 \\
0 & 0 & \lambda & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \lambda & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \lambda + 2\mu & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix} \quad (7)$$

The product of matrix $A_k$ and vector $q$ can be calculated as follows:

$$A_k = \rho^{-1}(T \cdot n) + \lambda(v \cdot n)I + \mu(n \otimes v + v \otimes n) \quad (8)$$

In the last equation $n$ is a unit vector directed along the $x$, $y$, or $z$ directions for matrices $A_1$, $A_2$, or $A_3$, respectively.

As we discussed above, the GCM approach is based on representing the solutions of the acoustic and/or elastic wave equations at later time as a linear combination of the displaced at a certain spatial step solutions at some previous time moment. This representation can be used to construct a direct time-stepping iterative algorithm of computing the wave fields at any time moment from the initial and boundary conditions. In order to develop this time-stepping formula, we represent matrices $A_k$ using their spectral decomposition. For example, for matrix $A_1$ we have:

$$A_1 = (\Omega_1)^{-1}A_1\Omega_1 \quad (9)$$

where $A_1$ is a $9 \times 9$ diagonal matrix, formed by the eigenvalues of matrix $A_1$; and $\Omega_1^{-1}$ is a $9 \times 9$ matrix formed by the corresponding eigenvectors. Note that, matrices $A_1$, $A_2$ and $A_3$ have the same set of eigenvalues:

$$\{c_p, -c_p, c_s, -c_s, c_s, -c_s, 0, 0, 0\}. \quad (10)$$

In the last formula, $c_p$ is a P-wave velocity being equal to $\left(\rho^{-1}(\lambda + 2\mu)\right)^{1/2}$ and $c_s$ is an S-wave velocity being equal to $(\rho^{-1}\mu)^{1/2}$.

Let us consider some direction $x$. We assume that the unit vector $n$ is directed along this direction, while the unit vectors $n_1$ and $n_2$ form a Cartesian basis together with $n$. We also introduce the following symmetric tensors of rank 2:

$$N_{i,j} = \frac{1}{2}(n_i \otimes n_j + n_j \otimes n_i), \quad (11)$$

where $\otimes$ denotes the tensor product of two vectors, the indices $i$ and $j$ vary from 0 to 2 in order to simplify the final formulas, and $n_0 = n$. 


The solution of equation (5), vector $q$, at the x, y, and z directions can be written as follows:

$$q(t + \tau, x, y, z) = \sum_{j=1}^{J} [X_{1,j}q(t, x - \Lambda_{1,j}\tau, y, z)],$$

$$q(t + \tau, x, y, z) = \sum_{j=1}^{J} [X_{2,j}q(t, x, y - \Lambda_{2,j}\tau, z)],$$

$$q(t + \tau, x, y, z) = \sum_{j=1}^{J} [X_{3,j}q(t, x, y, z - \Lambda_{3,j}\tau)].$$

(12)

3. The architecture of the software package

One of the results of this work is a software package that allows you to numerically solve the problems of wave propagation initiated by various dynamic loads on three-dimensional objects with a complex internal structure. The developed parallel version of the software package can significantly reduce the computation time, as well as increase the resolution of the models. The software package consists of three parts: a computational module based on the grid-characteristic method; input data preprocessing systems; post-processing systems for visualization and analysis of output data. Let’s consider each part of the software package in more detail.

The computing module has the following characteristics:

- the ability to calculate on three-dimensional parallelepiped grids and curvilinear structural grids;
- calculation on block geometry from a combination of grids, at the points of contacts of the grids they must be consistent;
- calculation using chimeric (superimposed grids);
- includes monotone grid-characteristic schemes of 1-4 orders of accuracy;
- has the ability to calculate both in one thread and on multi-core computing systems and on computing systems with distributed memory.

The architecture of the core of the computing complex was developed taking into account further enhancements of the capabilities of the software package. Preprocessing includes the following stages: setting the computational geometry; setting boundary conditions; setting the parameters of the initial disturbance region; setting the conditions at the interface between the media and at the boundaries of neighboring blocks; setting the mechanical characteristics of the medium; setting a computational algorithm; setting save parameters. Design geometry can be specified in several ways. First of all, you can specify the parameters of all computational grids directly in the configuration file, namely, the number of nodes along all axes, steps of the grids along all axes and their position in space. Curved meshes can be loaded from files, at the moment it is supported to load such meshes from the text version of the VTK format.

Also, the computational geometry can be prepared using the free package for pre- and post-processing scientific data Gmsh. At this stage, the import of block geometry from parallelepiped meshes is supported, the import of curved meshes is under development. In fig. 1 shows the process of creating the geometry of a high-rise building located in the enclosing rock mass.

After creation, the computational module independently breaks the geometry into blocks and in each block builds a computational grid with a given step. Contact boundaries and stitching conditions are set automatically. Also, the Gmsh capabilities allow you to specify the types of boundary conditions at the boundaries of the computational domains, and to designate areas with the same mechanical parameters of the environment. The use of this tool made it
Figure 1. Creation of computational geometry in the Gmsh program

possible to significantly expand the possibilities of specifying design geometries and to carry out modeling on complex geometries. At the moment, the following types of boundary conditions are implemented in the software package: a constant external force at the boundary is set; free surface condition; the external speed is set at the border; symmetry conditions for longitudinal and transverse elastic waves; absorption condition; demolition condition, i.e. the derivatives of all the functions determining the behavior of the environment vanish along the corresponding coordinate directions. The initial disturbance can be specified both in the form of a load versus time, and in setting the values of the sought functions at all nodes of the computational grid at the beginning of the calculation. There is a set of built-in initial conditions with different parameters, but import from external files is also supported. The software package allows you to automatically determine the locations of contact boundaries on geometries and, depending on the characteristics of the medium in neighboring blocks, set the condition for contact or stitching of neighboring areas. Depending on the type of partition, the following conditions can be set: conditions of free sliding; condition of complete adhesion; condition for stitching adjacent computational blocks. The setting of the values of the mechanical characteristics of the medium (density, velocity of longitudinal and shear waves) is carried out as a function of coordinates.

At the post-processing stage, graphic visualization and analysis of the calculated data are performed. The main output data format is VTK, and free packages that support this data format are used for 3D post-processing. The current version of the program supports work with the packages ParaView and Mayavi. In fig. 2 shows the process of processing the calculated data in the ParaView program. Also, the software package allows you to save data for quick analysis in the form of tables and one-dimensional graphs. This greatly simplifies the process of building seismograms and quick analysis of calculated data.
4. Parallelization

The considered software package is parallelized using MPI, OpenMP and CUDA technologies. Explicit grid methods are used in this work, i.e. the transition from one time step to the next occurs by traversing the entire computational grid and calculating the value at the new node based on the value at the original node and some of its vicinity. The mesh traversal is organized by two nested loops. The study of such an implementation has shown that with a sufficiently large size of the computational grid, a certain number of processor cache misses occurs, which leads to an increase in the program runtime. By changing the algorithm for traversing the computational grid, as well as through the use of additional memory arrays and forced data caching, it was possible to optimize the work with memory, which in some cases made it possible to speed up the operation of the algorithm on large computational grids [10].

To work in systems with distributed memory, the software package is parallelized using the MPI technology. When parallelizing, we used standard algorithms for decomposition of the computational domain and exchange of border cells. One of the main requirements for the algorithm is operation on a large number of computational cores (thousands) to provide an acceptable computation time. In Figure 3 shows the results of testing the scaling of the algorithm with an increase in the number of computational cores from 128 to 16384. In the test, a computational grid with a size of 1000 × 1000 × 1000 nodes was used [11].

The algorithm is also parallelized in shared memory systems using OpenMP technology. In this direction, work has just begun, and with an increase in the number of cores from 1 to 32, the algorithm shows an efficiency of about 60%. Such low rates are associated with insufficiently efficient work with memory, the computational cores have to access the memory areas of neighboring cores a lot, which slows down the calculation. Work is underway to improve the efficiency of work with the OpenMP technology [10].

The algorithm has also been parallelized on NVidia GPGPU using CUDA technology. A complete rewriting of a part of the computational module was required for this architecture. Currently, the speedup has been achieved about 80 times compared to a single CPU core using single precision numbers and 30 times using double precision numbers [12].

The software package also paid a lot of attention to parallelization in a distributed environment in the presence of many computational grids. Thus, an algorithm for the decomposition of multiblock meshes was proposed, which makes it possible to reduce the number of exchanges between processes [13].
5. Computational examples

Next, we will consider a number of examples calculated using this software package and demonstrating its capabilities. Figure 4 shows an example of the passage of a plane longitudinal wave front through a fluid-saturated oblique fracture.

The color reflects the longitudinal and transverse components of the wave. The transmitted and incident longitudinal waves, reflected and diffracted longitudinal waves (red) are clearly visible. Diffracted and reflected shear waves (blue) are also formed.

In fig. 5 shows an example of the passage of a plane wave through a porous medium. A
feature of the calculation is that each pore is clearly distinguished in the calculation algorithm. The next task is to calculate the seismic resistance of ground structures (Fig. 6, 7). Previous calculations were carried out taking into account the formation of cracks, but now we present the results for the shear fracture model. An earthquake is described as a single plane wave train falling from the Earth's interior perpendicular to the earth's surface. The structures are presented as monolithic concrete structures. In the first case, the structure is a domed structure (NPP dome) made of concrete and has a complex structure. The reactor is located in the center and is mounted on beams, along which seismic waves practically do not reach the reactor. The roof of the station has a dome shape also for greater seismic resistance. The height of the station is 30 m, the thickness of the walls is 4 m. It can be seen from the calculations that the destruction does not reach the reactor, which confirms the correctness of the chosen design.

The following is an example of calculating the destruction of a two-story residential building with a strip foundation (three-dimensional calculation). The task was posed with the participation of Indian specialists from CDAC, studying the seismic resistance of low-rise buildings in the foothills of the Himalayas. Wall thickness 40 cm, earthquake magnitude 0.1 m/s, 5 times less than in the previous calculation. Two series of calculations were carried out for different strength moduli (1.0 MPa and 1.5 MPa). In fig. 7 shows a comparison of the fracture
Figure 7. Seismic resistance of a two-story residential building pattern without taking into account the corrections caused by the destroyed areas (left), and taking into account (right).

Conclusions
The paper describes a software package for modeling the propagation of dynamic wave disturbances in heterogeneous media. A feature of the algorithms is the explicit identification of inhomogeneities. The software package is parallelized in systems with shared and distributed memory, as well as on graphics coprocessors. Examples of calculations are given that show the broad capabilities of the developed software package.

Acknowledgements
This work was supported by RFBR (Russian Foundation for Basic Research), the Government of Krasnoyarsk krai and Krasnoyarsk Regional Fund for Supporting Scientific and Technological Activities Project № 20-48-242923 p_mk_Krasnoyarsk, Research of separation and purification of gas mixtures by methods of kinetic theory in new generation porous nanomaterials.

References
[1] Stognii P, Khokhlov N and Petrov I 2021 Journal of Physics: Conference Series 1715 012058 DOI: 10.1088/1742-6596/1715/1/012058
[2] Kozhemyachenko A A, Petrov I B, Favorskaya A V and Khokhlov N I 2020 Comput. Math. Math. Phys 60 1539–1554 DOI: 10.1134/S0965542520090110
[3] Golubev V I, Shevchenko A V and Petrov I B 2021 Journal of Physics: Conference Series 1715 012056 DOI: 10.1088/1742-6596/1715/1/012056
[4] Stognii P V, Khokhlov N I and Petrov I B 2020 Numer. Anal. Appl. 13 271–281 DOI: 10.1134/S1995423920030076
[5] Favorskaya A V and Khokhlov N I 2020 *Procedia Computer Science* **176** 2556–2565 ISSN 1877-0509 Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 24th International Conference KES2020 DOI: 10.1016/j.procs.2020.09.317

[6] Beklemysheva K A and Petrov I B 2020 *Lobachevskii J. Math.* **41** 2638–2647 DOI: 10.1134/S1995080220120082

[7] LeVeque R J 2002 *Finite Volume Methods for Hyperbolic Problems* Cambridge Texts in Applied Mathematics (Cambridge University Press) DOI: 10.1017/CBO9780511791253

[8] Favorskaya A V, Zhdanov M S, Khokhlov N I and Petrov I B 2018 *Geophysical Prospecting* **66** 1485–1502 DOI: 10.1111/1365-2478.12639

[9] Golubev V I, Petrov I B and Khokhlov N I 2013 *Comp. Math. and Math. Phys.* **53** 1523–1533

[10] Furgailo V, Ivanov A and Khokhlov N 2019 *2019 Ivannikov Memorial Workshop (IVMEM)* pp 79–85 DOI: 10.1109/IVMEM.2019.00019

[11] Ivanov A M and Khokhlov N I 2018 *Comput. Res. Model.* **10** 667–678 DOI: 10.20537/2076-7633-2018-10-5-667-678

[12] Khokhlov N, Ivanov A, Zhdanov M, Petrov I and Ryabinkin E 2016 *Distributed Computer and Communication Networks* ed Vishnevskiy V M, Samosylov K E and Kozyrev D V (Cham: Springer International Publishing) pp 577–588 ISBN 978-3-319-51917-3 DOI: 10.1007/978-3-319-51917-3_9

[13] Fofanov V and Khokhlov N 2020 *Supercomputing* ed Voevodin V and Sobolev S (Cham: Springer International Publishing) pp 63–73 ISBN 978-3-030-64616-5 DOI: 10.1007/978-3-030-64616-5_6