Hydrodynamic modeling of self-gravitating astrophysical objects on tetrahedral meshes

Igor Kulikov\textsuperscript{1}, Eduard Vorobyov\textsuperscript{2}, Igor Chernykh\textsuperscript{1} and Vardan Elbakyan\textsuperscript{3}

\textsuperscript{1}Institute of Computational Mathematics and Mathematical Geophysics SB RAS, Lavrentieva ave., 6, Novosibirsk, 630090 Russia
\textsuperscript{2}Department of Astrophysics, University of Vienna, Tuerkenschanzstrasse, 17, Vienna, 1180 Austria
\textsuperscript{3}Research Institute of Physics, Southern Federal Univeristy, Stachki ave., 194, Rostov-on-Don, 344090 Russia

Abstract. The paper proposes a new numerical method for solving the equations of gravitational hydrodynamics on a tetrahedral mesh. The proposed numerical method is focused on modeling the evolution of astrophysical objects of spherical shape, which is appropriate for gravitational collapse and star formation, and also for supernova explosion. The construction of tetrahedral grids is carried out in three stages. At the first stage, a geodesic grid methodology is used to construct a triangular grid on the surface of the sphere, which encompasses the computational domain. At the second stage, the resulting triangular mesh is serialized from the surface of the sphere to its center, and at the third stage, the obtained prisms are divided into tetrahedra. This approach allows us to simulate spherical objects without singularities that occur when using spherical or cylindrical coordinates. The paper describes numerical methods for solving the equations of hydrodynamics and the Poisson equation. Numerical examples are given that verify the developed numerical methods.

1. Introduction
Mathematical modeling is one of the most important theoretical tools for studying astrophysical objects. The main trends in hydrodynamic modeling are the Lagrangian method of smoothed particles \cite{1, 2} and Euler methods based on adaptive grids \cite{3}. Over the past ten years, numerical methods based on moving grids \cite{4, 5}, as well as on a combination of Lagrangian and Euler approaches, grid-based \cite{6, 7} or meshless \cite{8}, were actively developed. The main trend of all these approaches is the adaptation of the numerical discretization to the physical features of the problem.

We present an original approach to address such features based on tetrahedral meshes. Given our interest in star formation problems, the construction of numerical meshes is based on the use of geodesic grids \cite{9}. The use of such grids allows us to construct efficient parallel numerical solution methods \cite{10, 11, 12}, including methods of high-order accuracy \cite{13}.

The second section describes the procedure for constructing tetrahedral meshes for modeling spherical objects. The third section briefly describes numerical methods for solving hydrodynamic equations on tetrahedral meshes and a method for solving the Poisson equation. The fourth section is devoted to numerical examples. In the fifth section, a conclusion is formulated.
2. Tetrahedral meshes

The construction of a tetrahedral mesh is performed in three steps:

(i) Construction of a geodesical mesh on the surface of a sphere that encompasses the computational region.

(ii) Serialization of the geodesical mesh from the surface to the center of the computational domain.

(iii) Splitting of the resulting prisms into 17 tetrahedrons.

The scheme for constructing the tetrahedral mesh is shown in Fig. 1. The green color highlights...
The main nodes of the prism, the blue color indicates the nodes located at the faces of the prism, the pink color indicates the node that is placed in the center of the prism.

3. Numerical method

3.1. The method for solving the hydrodynamic equations

We will consider the equations of hydrodynamics written in the form of conservation laws for mass, momentum, and total energy:

\[
\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho u_x u_x + p \\ \rho u_x u_y \\ \rho u_x u_z \\ \rho u_x u_x \\ \rho E + p \end{pmatrix} + \frac{\partial}{\partial y} \begin{pmatrix} \rho u_y u_x + p \\ \rho u_y u_y \\ \rho u_y u_z \\ \rho u_y u_y \\ \rho E + p \end{pmatrix} + \frac{\partial}{\partial z} \begin{pmatrix} \rho u_z u_x + p \\ \rho u_z u_y + p \\ \rho u_z u_z \\ \rho u_z u_z \\ \rho E + p \end{pmatrix} = 0,
\]

where \( \rho \) is the gas volume density, \( \vec{u} = (u_x, u_y, u_z) \) the gas velocity, \( p \) the gas pressure, \( E = p/(\gamma-1)+\rho\vec{u}^2/2 \) the total mechanical energy of gas, \( \gamma \) the adiabatic index. The conservation laws in the vector form are:

\[
\frac{\partial \vec{u}}{\partial t} + \frac{\partial f(\vec{u})}{\partial x} + \frac{\partial g(\vec{u})}{\partial y} + \frac{\partial h(\vec{u})}{\partial z} = 0,
\]

where \( \vec{u} \) is the vector of conservative variables, functions \( f(\vec{u}), g(\vec{u}), h(\vec{u}) \) the fluxes in the corresponding directions. Additionally, we will consider the vector of physical variables \( q(\vec{u}) = (\rho, u_x, u_y, u_z, p) \). We assume that a simple procedure exists that can recover \( q(\vec{u}) \) from the vector of conservative variables.

Let us consider an arbitrary tetrahedral cell with the ordinal number 1. For describing a set of neighboring cells we will use index \( j \). We then define the normal \( n_{ij} \) of the cell \( i \) in the direction of cells \( j \) for the known vector \( \vec{M}_i, \vec{M}_{ij} \), \( M_i \) the center of the sphere inscribed in the tetrahedron \( i \), \( M_{ij} \) the center of the circle inscribed in the face between cells \( i \) and \( j \). The area of faces and the volume of cells will also be necessary in the calculations. The volume of tetrahedron \( V_i \) and triangles \( S_{ij} \) can easily be written. Integrating the equations and considering the finite volume \( V_i \) with the boundary \( \Gamma_i = \bigcup_j S_{ij} \), we obtain:

\[
\int_{V_i} \frac{\partial u}{\partial t} \, dV + \int_{V_i} \left( \frac{\partial f(u)}{\partial x} + \frac{\partial g(u)}{\partial y} + \frac{\partial h(u)}{\partial z} \right) \, dV = 0,
\]

using the Gauss theorem, the integral equations can be written as

\[
V_i \frac{du}{dt} + \oint_{\Gamma_i} (fn^x + gn^y + hn^z) \, dS = 0.
\]

The finite-volume scheme has the form:

\[
\frac{u_i^{n+1} - u_i^n}{\tau} + \sum_j \frac{S_{ij}}{V_i} \left( F_{ij}n_i^x + G_{ij}n_i^y + H_{ij}n_i^z \right) = 0,
\]

where \( F_{ij}, G_{ij}, H_{ij} \) is the fluxes in the Riemann problem \( \mathcal{R} = (w, u^L, u^R) \) for equations:

\[
\frac{\partial u}{\partial t} + \frac{\partial w(u)}{\partial x} = 0,
\]
with the initial conditions \( u(x, t) = u^L \) for \( x < 0 \) \( u(x, t) = u^R \) for \( x \geq 0 \). To solve the Riemann problem, we will use the scheme of Rusanov [14, 15]:

\[
R \left( w, u^L, u^R \right) = \frac{w(u^L) + w(u^R)}{2} + \frac{\left| \frac{\partial w}{\partial u} \right|}{2} \left( u^L - u^R \right).
\]

To determine the timestep \( \tau \), we define the diameters of spheres \( d_i \) that are inscribed in every grid cell. Then the timestep can be calculated as:

\[
\tau = \min_i \left( \frac{d_i}{\max \left( \left| \frac{\partial F(u_i)}{\partial u}, \frac{\partial G(u_i)}{\partial u}, \frac{\partial H(u_i)}{\partial u} \right| \right)} \right) \times CFL,
\]

where \( CFL < 1 \) ia the the Courant number. To increase the order of accuracy, we employ the apprach analogous to that described in [16].

3.2. Method for solving the Poisson equation

The gravitational potential in equations of hydrodynamics is found by solving for the Poisson equation:

\[
\nabla^2 \Phi = 4\pi G\rho.
\]

The computed potential is taken into account in the right-hand-side of equations of hydrodynamics:

\[
\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \end{pmatrix} = - \begin{pmatrix} 0 \\ \rho \nabla \Phi_x \\ \rho \nabla \Phi_y \\ \rho \nabla \Phi_z \\ \rho (\vec{u}, \nabla \Phi) \end{pmatrix}.
\]

To solve the Poisson equations, we consider the potential determined at the nodes of the computational grid, and employ the weak-set finite element method for discretization of the Laplace operator. Let us consider an arbitrary tetrahedon \( ABCD \), the vertices of which \( A = (A_x, A_y, A_z), B = (B_x, B_y, B_z), C = (C_x, C_y, C_z) \) and \( D = (D_x, D_y, D_z) \), respectively. We define four basis functions, each of which is unique in only one node, and in the others they are set equal to zero:

\[
\psi_i(x, y, z) = a_i x + b_i y + c_i z + d_i,
\]

where \( i = A, B, C, D \). To determine the coefficients \( a_i, b_i, c_i, d_i \), the following matrix equation needs to be solved:

\[
\begin{pmatrix}
A_x & A_y & A_z & 1 \\
B_x & B_y & B_z & 1 \\
C_x & C_y & C_z & 1 \\
D_x & D_y & D_z & 1
\end{pmatrix}
\begin{pmatrix}
a_A & a_B & a_C & a_D \\
b_A & b_B & b_C & b_D \\
c_A & c_B & c_C & c_D \\
d_A & d_B & d_C & d_D
\end{pmatrix}
= \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}.
\]

This system can be analytically solved using the Cramer method. In this case, the gradient of each basis function in the tetrahedron has a simple analytical form:

\[
\nabla \psi_i = (a_i, b_i, c_i),
\]

the sum of which allows us to build a potential gradient in a tetrahedral cell:

\[
\nabla \Phi = \begin{pmatrix}
\Phi_{AA} + \Phi_{BB} + \Phi_{CC} + \Phi_{DD} \\
\Phi_{AB} + \Phi_{BB} + \Phi_{CB} + \Phi_{DB} \\
\Phi_{AC} + \Phi_{BC} + \Phi_{CC} + \Phi_{DC}
\end{pmatrix},
\]

where

\[
\Phi_{AB} = \int_{A}^{B} \phi \, ds,
\]

\[
\Phi_{BC} = \int_{B}^{C} \phi \, ds,
\]

\[
\Phi_{CD} = \int_{C}^{D} \phi \, ds,
\]

\[
\Phi_{DA} = \int_{D}^{A} \phi \, ds.
\]
where $\Phi_i$ are the values of the potential in the corresponding nodes. When constructing a high-order accuracy scheme for solving hydrodynamic equations, we projected the density function into nodes, therefore, like the potential, density is also considered at the nodes of the tetrahedron.

In conclusion, we provide the formulas for the stiffness matrix $S$ and the mass matrix $M$ for the finite element formulation for the tetrahedron under consideration:

$$
S = -V_{ABCD} \begin{pmatrix}
(\nabla \psi_A, \nabla \psi_A) & (\nabla \psi_A, \nabla \psi_B) & (\nabla \psi_A, \nabla \psi_C) & (\nabla \psi_A, \nabla \psi_D) \\
(\nabla \psi_B, \nabla \psi_A) & (\nabla \psi_B, \nabla \psi_B) & (\nabla \psi_B, \nabla \psi_C) & (\nabla \psi_B, \nabla \psi_D) \\
(\nabla \psi_C, \nabla \psi_A) & (\nabla \psi_C, \nabla \psi_B) & (\nabla \psi_C, \nabla \psi_C) & (\nabla \psi_C, \nabla \psi_D) \\
(\nabla \psi_D, \nabla \psi_A) & (\nabla \psi_D, \nabla \psi_B) & (\nabla \psi_D, \nabla \psi_C) & (\nabla \psi_D, \nabla \psi_D)
\end{pmatrix}
$$

$$
M = \frac{V_{ABCD}}{20} \begin{pmatrix}
2 & 1 & 1 & 1 \\
1 & 2 & 1 & 1 \\
1 & 1 & 2 & 1 \\
1 & 1 & 1 & 2
\end{pmatrix}.
$$

At the outer boundary of the region, a fundamental solution of the Laplace equations in the form of boundary conditions of the first kind is used. To solve the resulting sparse system of equations, the conjugate gradient method is used.

3.3. The use of moving meshes

Let us define an analytic smooth function that defines the movement of the computational domain $\vec{w} = (w_x, w_y, w_z)$. Smoothness is needed to eliminate the possible effect of tetrahydron degeneracy. The following function can be considered $\frac{d\vec{w}}{dt} \approx \nabla \Phi$. The equations of hydrodynamics can then be written as

$$
\frac{\partial}{\partial t} \left( \begin{pmatrix} \rho \\ \rho \vec{u} \\ \rho E \end{pmatrix} \right) + \nabla \cdot \left( \begin{pmatrix} \rho (\vec{u} + \vec{w} - \vec{w}) \\ \rho \vec{u}(\vec{u} + \vec{w} - \vec{w}) + \vec{P} \\ [\rho E + p](\vec{u} + \vec{w} - \vec{w}) \end{pmatrix} \right) = 0,
$$

where $\vec{P}$ is the diagonal pressure tensor. After introducing the vector of effective gas velocity with respect to the moving mesh $\vec{v} = \vec{u} - \vec{w}$, the equations can be rewritten in the form of two systems:

$$
\frac{\partial}{\partial t} \left( \begin{pmatrix} \rho \\ \rho \vec{u} \\ \rho E \end{pmatrix} \right) + \nabla \cdot \left( \begin{pmatrix} \rho \vec{v} \\ \rho \vec{u} \vec{v} + \vec{P} \\ [\rho E + p] \vec{v} \end{pmatrix} \right) = 0,
$$

$$
\frac{\partial}{\partial t} \left( \begin{pmatrix} \rho \\ \rho \vec{u} \\ \rho E \end{pmatrix} \right) + \nabla \cdot \left( \begin{pmatrix} \rho \vec{w} \\ \rho \vec{u} \vec{w} \\ \rho E \vec{w} \end{pmatrix} \right) = 0.
$$

The first system can be solved according to the scheme provided earlier with the corresponding modification of the fluxes. The solution of the second system is not actually carried out, the grid nodes move with the conservation of mass, angular momentum, and the total energy in each computational cell. In the current implementation, we retained the possibility of including the movement of the computational grid, but in computational experiments this possibility was not used. In the future, we will study in detail the algorithms for the grid movement for each problem to be solved.
3.4. The entropy equation
As an additional equation for writing an overdetermined system of hydrodynamic equations and for regularizing the numerical solution, we can use the equation for entropy, also written in the conservative form. This approach was effectively used in [12, 17, 18, 19] to achieve a non-negative value of pressure in regions with a high Mach number and to guarantee a non-decreasing entropy in the numerical method. The advantages and disadvantages of using the entropy equation are also discussed in detail in [13]. In the current implementation, we have retained the possibility of including the entropy equation in the numerical model. In computational experiments, this equation was not used.

4. Numerical examples
We consider four test problems as numerical examples: the problem of discontinuity decay to test the ability of the method to treat all hydrodynamic waves, the Sedov problem for testing the treatment of strong shock waves, the problem of solving Poisson’s equation with differentiable functions to check the convergence degree of the method, and the collapse of a gas cloud to test the method in a complete mathematical model.

4.1. Decay of discontinuity
Let us consider the dynamics of gas on a segment \([0; 1]\) till a time instance of \(t = 0.2\). At \(x_0 = 0.5\) the gas distribution has a discontinuity. The gas parameters on the left are: \(p^L = 2\) is the pressure, \(\rho^L = 2\) is the density, and \(v^L = 0\) is the velocity. On the right from the discontinuity the gas parameters are: \(p^R = 1\), \(\rho^R = 1\), and \(v^R = 0\). The adiabatic index is \(\gamma = 1.4\). We set the non-reflecting boundary conditions on both sides. The results are presented in Fig. 2. 200 grid cells were used for this test.

The purpose of this test is to demonstrate the correct computation of contact discontinuity, rarefaction waves, and shock waves. Most methods for solving gas dynamics equations give either oscillations or diffusion on the position of the shock. The author’s method of high order in accuracy reveals a small (by two cells) smearing of the solution at the shock wave region.

4.2. The Sedov test for a point explosion
Sedov’s problem of a point explosion in astrophysics is designed to simulate a central supernova explosion. To model the point explosion problem, we consider a region \([-0.5; 0.5]^3\), the adiabatic exponent \(\gamma = 5/3\), the initial density in the region \(\rho_0 = 1\), and the initial pressure \(p_0 = 10^{-5}\). At a time instance of \(t = 0\), the internal energy \(E_0 = 0.6\) is released. The explosion region is limited by the radius \(r_{central} = 0.01\). The simulated profile of density and angular momentum at \(t = 0.05\) is shown in Fig. 3.

The Sedov point explosion problem is a standard test that checks the ability of a numerical method to reproduce strong shock waves with large Mach numbers. The sound speed of the background medium is negligible; therefore, the Mach number reaches a value \(M \approx 1500\). Our numerical method reproduces well the position of the shock wave, as well as the radial profile of the density and momentum.

4.3. Solution of the Poisson equation
The solution of the Poisson equation was studied by refining the computational grid and using the known analytical solution of the gravitational potential and the density function:

\[
\Phi(r) = \begin{cases} 
\frac{4\pi}{15} r^5 - \frac{3\pi}{5} r^4 + \frac{2\pi}{3} r^3 - \frac{3\pi}{2}, & r \leq 1, \\
-\frac{4\pi}{15r}, & r > 1.
\end{cases}
\]

\[
\rho(r) = \begin{cases} 
2r^3 - 3r^2 + 1, & r \leq 1, \\
0, & r > 1.
\end{cases}
\]
Figure 2. Decay of an arbitrary discontinuity: density (top left), pressure (top right), velocity (bottom left), and internal energy (bottom right). The thin line shows the exact solution, circles indicate the numerical solution.

| Number of cells | Relative residuals |
|-----------------|-------------------|
| 32 720          | 10.64E-3          |
| 266 880         | 3.35E-3           |
| 2 155 520       | 0.82E-3           |

Table 1. Euclidean norm of the deviation of the solution when refining the mesh.

Table 1 provides the values of relative residuals (errors) during the mesh refinement. As can be seen from the Table, the second order of convergence is achieved. The scheme has the second-order of convergence.

4.4. Collapse of a gaseous cloud
Let us consider the gravitational collapse of an initially static gaseous cloud with pressure $p = \rho/10$, where the density profile has the form:

$$\rho(r) = \begin{cases} 
2r^3 - 3r^2 + 1, & r \leq 1, \\
0, & r > 1.
\end{cases}$$

The simulated gas density profile at time $t = 0.5$ is shown in Fig. 4. As can be seen from
Figure 3. Density (left) and angular momentum (right) obtained by numerically solving the Sedov problem of a point explosion. The solid line shows the exact solution.

Figure 4. Gas density. The solid line shows the initial density profile, the circles show the density of the collapsing gas.

the Figure, the maximum density value increases by a factor of six, while most of the mass is concentrated in a sphere that is twice smaller than the initial distribution.

5. Conclusion
A new numerical method for solving the equations of gravitational hydrodynamics on tetrahedral grids is developed. A procedure for constructing tetrahedral grids for modeling spherical astrophysical objects based on geodetic grids is proposed. The results of computational experiments for verification of the developed numerical method are presented.

Acknowledgements
The reported study was funded by RFBR and FWF according to the research project 19-51-14002 (RFBR) and I4311 (FWF).
References

[1] Gingold R.A., Monaghan J.J. Smoothed particle hydrodynamics - Theory and application to non-spherical stars // Monthly Notices of the Royal Astronomical Society. – 1977. – V. 181. – P. 375-389.

[2] Lucy L.B. A numerical approach to the testing of the fission hypothesis // The Astrophysical Journal. – 1977. – V. 82. – P. 1013-1024.

[3] O’Shea B., Bryan G., Bordner J., Norman M., Abel T., Harkness R., Kritsuk A. Adaptive Mesh Refinement - Theory and Applications // Lectures Notes of Computer Science Engineering. – 2005. – V. 41. – P. 341-350.

[4] Springel V. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh // Monthly Notices of the Royal Astronomical Society. – 2010. – V. 401. – P. 791-851.

[5] Weinberger R., Springel V., Pakmor R. The AREPO Public Code Release // The Astrophysical Journal Supplement Series. – 2020. – V. 248. – Article Number 32.

[6] Murphy J., Burrows A. BETHE-Hydro: An Arbitrary Lagrangian-Eulerian Multidimensional Hydrodynamics Code for Astrophysical Simulations // The Astrophysical Journal Supplement Series. – 2008. – V. 179. – P. 209-241.

[7] Ardeljan N., Bisnovatyi-Kogan G., Moiseenko S. An implicit Lagrangian code for the treatment of nonstationary problems in rotating astrophysical bodies // Astronomy and Astrophysics Supplement. – 1996. – V. 115. – P. 573-594.

[8] Hopkins P. A new class of accurate, mesh-free hydrodynamic simulation methods // Monthly Notices of the Royal Astronomical Society. – 2015. – V. 450. – P. 53-110.

[9] Balsara D., Florinski V., Garain S., Subramanian S., Gurski K. Efficient, divergence-free, high-order MHD on 3D spherical meshes with optimal geodesic meshing // Monthly Notices of the Royal Astronomical Society. – 2019. – V. 487. – P. 1283-1314.

[10] Kulikov I. GPUPEGAS: A New GPU-accelerated Hydrodynamic Code for Numerical Simulations of Interacting Galaxies // The Astrophysical Journal Supplement Series. – 2014. – V. 214. – Article Number 12.

[11] Kulikov I.M., Chernykh I.G., Snytnikov A.V., Glinskiy B.M., Tutukov A.V. AstroPhi: A code for complex simulation of dynamics of astrophysical objects using hybrid supercomputers // Computer Physics Communications. – 2015. – V. 186. – P. 71-80.

[12] Kulikov I., Chernykh I., Tutukov A. A New Hydrodynamic Code with Explicit Vectorization Instructions Optimizations that Is Dedicated to the Numerical Simulation of Astrophysical Gas Flow. I. Numerical Method, Tests, and Model Problems // The Astrophysical Journal Supplement Series. – 2019. – V. 243. – Article Number 4.

[13] Kulikov I., Vorobyov E. Using the PPML approach for constructing a low-dissipation, operator-splitting scheme for numerical simulations of hydrodynamic flows // Journal of Computational Physics. – 2016. – V. 317. – P. 318-346.

[14] Kulikov I.M., Chernykh I.G., Glinskiy B.M., Protasov V.A. An Efficient Optimization of Hll Method for the Second Generation of Intel Xeon Phi Processor // Lobachevskii Journal of Mathematics. – 2018. – V. 39, I. 4. – P. 543-551.

[15] Kulikov I.M., Chernykh I.G., Tutukov A.V. A New Parallel Intel Xeon Phi Hydrodynamics Code for Massively Parallel Supercomputers // Lobachevskii Journal of Mathematics. – 2018. – V. 39, I. 9. – P. 1207-1216.

[16] Chen G., Tang H., Zhang P. Second-Order Accurate Godunov Scheme for Multicomponent Flows on Moving Triangular Meshes // Journal of Scientific Computing. – 2008. – V. 34. – P. 64-86.

[17] Balsara D., Spicer D. Maintaining Pressure Positivity in Magnetohydrodynamic Simulations // Journal of Computational Physics. – 1999. – V. 148. – P. 133-148.

[18] Springel V., Hernquist L. Cosmological smoothed particle hydrodynamics simulations: the entropy equation // Monthly Notices of the Royal Astronomical Society. – 2002. – V. 333. – P. 649-664.

[19] Godunov S., Kulikov I. Computation of Discontinuous Solutions of Fluid Dynamics Equations with Entropy Nondecrease Guarantee // Computational Mathematics and Mathematical Physics. – 2014. – V. 54. – P. 1012-1024.