

Modeling and Visualization of Sedimentation in a Viscous Incompressible Fluid

V. A. Galkin^{1,A,B}, A.O. Dubovik^{2,A,B}, A.D. Smorodinov^{3,A,B}

^A Surgut branch of SRISA, Surgut, Russia

^B Surgut State University, Surgut, Russia

¹ ORCID: 0000-0002-9721-4026, val-gal@yandex.ru

² ORCID: 0000-0002-4158-9646, alldubovik@gmail.com

³ ORCID: 0000-0002-9324-1844, sachenka_1998@mail.ru

Abstract

The article considers the equation of hydrodynamics with a sliding condition at the boundary and the motion of an inertialess non-diffusing impurity in a viscous incompressible liquid in a cylindrical region. Diffusion processes in dispersed systems are investigated – the effect of parachuting a falling large particle in a liquid-filled medium. The result of a numerical and analytical solution of a system of equations consisting of the Navier-Stokes equation for an incompressible fluid and the convection-diffusion equation is presented. It is shown that in order to calculate the trajectory of an impurity particle, it is necessary to solve the Cauchy problem in three-dimensional space with given initial conditions for each impurity particle. A series of computational experiments has been carried out to simulate the dynamics of a heavy impurity in an incompressible liquid. The initial conditions for the Cauchy problem are set based on a black-and-white image, where a black pixel is considered an impurity particle. The developed software has been tested to determine the degree of confidence in the calculations performed by solving the initial problem in reverse order, for which the initial conditions for the Cauchy problem were chosen equal to the value at the last step of the initial problem. A visual and quantitative analysis of the results obtained is presented based on the developed software package using the MathGL library for the C++ programming language.

Keywords: mathematical modeling, hydrodynamics, Navier-Stokes equation, sedimentation, Cauchy problem, Runge-Kutta method.

Let us consider a system of equations of hydrodynamics with a sliding condition at the boundary and the motion of an inertialess non-diffusing impurity:

$$\left\{ \begin{array}{l} \frac{\partial \vec{V}}{\partial t} + (\vec{V} \nabla, \vec{V}) + \frac{1}{\bar{\rho}} \nabla P = \varepsilon^2 \Delta \vec{V}, \\ \operatorname{div} \vec{V} = 0, \\ (\vec{V}, \vec{n})|_{\partial D} = 0, \\ \frac{\partial N}{\partial t} + (\vec{V} \cdot \nabla) N = -\operatorname{div} \frac{\vec{F}}{h}. \end{array} \right. \quad (1)$$

Equations (1) are investigated in the three-dimensional domain $D \subset \mathbb{R}^3 = \{\vec{X} = (x_1, x_2, x_3)\}$, where $\vec{V}(\vec{X}, t)$ – the velocity field of the fluid flow, which obeys the condition of non-flow at the boundary of the region ∂D , and $P(x, t)$ – the pressure field in the same three-dimensional region; $\varepsilon^2, \bar{\rho}$ – are constants that characterize the viscosity and density of the liquid, N – the concentration of the impurity in the flow field. \vec{F} – an external force acting on particles, h – the coefficient of resistance to particle motion. In the future, gravity is considered as an external force.

$\vec{n}(\vec{X})$ – the vector of the normal to the surface ∂D at a point on the surface $x \in \partial D$.

The article presents the result of a numerically analytical solution of the system of equations (1) consisting of the Navier-Stokes equation for an incompressible fluid and the convection-diffusion equation and is a continuation of the approaches and visualization methods developed in [1]. In the framework of model (1), diffusion processes in dispersed systems are investigated, namely, the effect of parachuting a falling large particle in a liquid-filled medium. This effect is one of the causes of particle coagulation [2]. Large particles and the liquid flow surrounding them have identical physical parameters: viscosity, density, temperature, but differ in size and, therefore, have different masses.

Convective diffusion problems are an intensive field of study [3-7], however, as a rule, they are considered in isolation from the equations describing the dynamics of the flow field, which is considered to be set [3-5]. In [5], a numerical and analytical solution to the problem of migration of impurities in the atmosphere and water medium is considered. In [6], a stationary problem consisting of a convection-diffusion equation and an equation for a flow field with a nonlinear reaction coefficient is considered, and in [7], an exclusively numerical solution using foreign Ansys Fluent software is considered. In this paper, we study the nonstationary convection-diffusion problem describing the dynamics of a heavy impurity in the incompressible fluid flow field, using the class of exact solutions of the Navier-Stokes equation for an incompressible fluid found by the authors and the developed software for calculating and visualizing the impurity deposition process.

According to [8] field $\vec{V}(\vec{X}, t)$, given by formula (2), and the scalar field $P(\vec{X}, t)$ (3), they are the solution of a system of Navier-Stokes equations for the flow of a viscous incompressible liquid in a cylinder:

$\Pi_k = \{\vec{X} \in \mathbb{R}^3 : 0 < r(\vec{X}) < \rho_k \sqrt{2}, 0 < x_3 < 2\pi\sqrt{2}\}$, where $r(\vec{X}) = \sqrt{x_1^2 + x_2^2}$, and $\rho_k > 0$, $k \in \mathbb{N}$ – the roots of the derivative of the Bessel function of the first kind of zero order, ordered in ascending order.

$$\vec{V}(\vec{X}, t) = \vec{U}(\vec{X}) \exp(-\varepsilon^2 t), \quad (2)$$

$$P(\vec{X}, t) = -\frac{\bar{\rho}}{2} \vec{V}^2(\vec{X}, t) + \alpha(t), \quad (3)$$

$$\vec{U}(\vec{X}) = \frac{J'_0\left(\frac{r}{\sqrt{2}}\right)}{r\sqrt{2}} \begin{bmatrix} x_2 \sin\left(\frac{x_3}{\sqrt{2}}\right) + \frac{x_1}{\sqrt{2}} \cos\left(\frac{x_3}{\sqrt{2}}\right) \\ -x_1 \sin\left(\frac{x_3}{\sqrt{2}}\right) + \frac{x_2}{\sqrt{2}} \cos\left(\frac{x_3}{\sqrt{2}}\right) \\ 0 \end{bmatrix} + \frac{1}{2} J_0\left(\frac{r}{\sqrt{2}}\right) \begin{bmatrix} 0 \\ 0 \\ \sin\left(\frac{x_3}{\sqrt{2}}\right) \end{bmatrix}, \quad (4)$$

where

J_0 – the Bessel function of the 0th order,

J'_0 – the derivative of the Bessel function of the 0th order,

$\alpha(t)$ – an arbitrary function of time t .

First, equation (4) is considered without taking into account the action of an external volumetric force. In this case, the velocity vector of the impurity coincides with the velocity vector of the liquid \vec{V} . Therefore, to calculate the trajectory of an impurity particle, it is necessary to solve the Cauchy problem in three-dimensional space with given initial conditions for each impurity particle.

The developed software solves the Cauchy problem for calculating and visualizing the impurity deposition process, using the Runge-Kutta method of the 4th order of accuracy, which is a classical numerical method for solving this problem [9, 10]. To build the visualization, the MathGL library [11] for the C++ programming language was used, a computer based on the AMD Ryzen Threadripper 2990Wx 32-Core Processor, with 32 GB of RAM, was used as computing stations to solve the problem. To unlock the potential of the processor and accelerate calculations, OpenMP technology was used [12] with parallelization of calculations on 64-threads.

A series of computational experiments has been carried out to simulate the dynamics of a heavy impurity in an incompressible liquid. The initial position of the impurity is given by the expression

$$N_k(t = 0, \vec{X}) = (x_{1,k}^0, x_{2,k}^0, x_{3,k}^0).$$

The k index indicates the number of the impurity particle for which the initial position is set. The initial position of each impurity particle is determined by the following rule:

1. Consider an arbitrary, binary (two-color) black-and-white image with a size of no more than 300 pixels by 300 pixels, on which the white color will serve as the background and occupy from 50% of the image, the black color is the contour of an arbitrary image, for example, consider Fig. 1. The limitation on the image size is due to the capabilities of the computer.

2. Let's denote the size of the picture in pixels by w, h – width and height. Consider each pixel of the image $P_{i,j}$, where P – the pixel value (1 is white, 0 is black), and the pair (i, j) – pixel coordinates in the left rectangular coordinate system, the origin of which is the upper-left corner of the image.

3. If the pixel $P_{i,j}$ if the value is 0 (i.e. it is black), then this pixel will be used as the initial condition for the Cauchy problem after processing using the following formulas:

$$x_{1,k}^0 = \frac{i}{w*0.1} - Sq_0, x_{2,k}^0 = \frac{j}{h*0.1} - Sq_0,$$

where Sq_0 – the size of the area in which the impurity is to be placed (in the article under consideration $Sq_0 = 5$).

The coordinate $x_{2,k}^0 = 0.1$ for each impurity particle, its value corresponds to the location of the particle above the earth's surface.

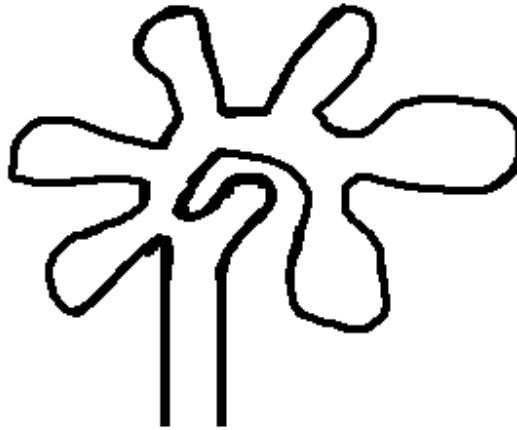


Fig. 1. The image used to form the initial impurity field

The main characteristics of computational experiments: $t_{start} = 0; t_{end} = 400$; *time step for the Runge – Kutta algorithm*: $dt = 10^{-5}$, time step to save the results $dt = 0.5$ (every 50,000th image was saved), the viscosity coefficient $\varepsilon^2 = 0.05^2$.

The results of mathematical modeling with the specified parameters are shown in Fig. 2.

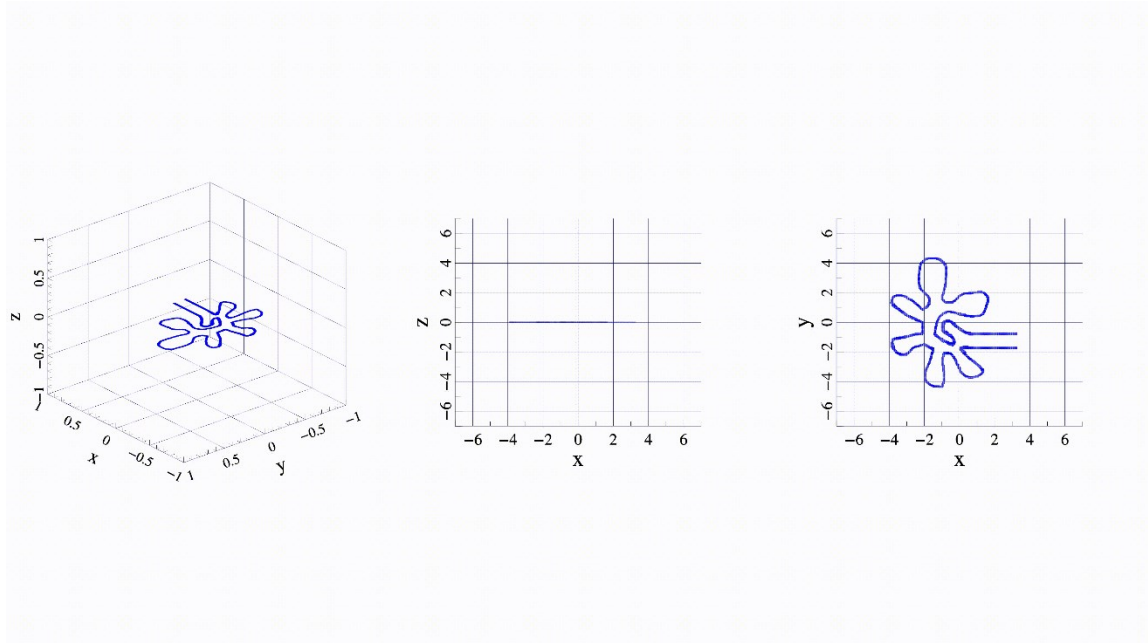


Fig. 2. Results of mathematical modeling for the 1st computational experiment

For software testing, the inverse problem is considered, Fig. 3. In this case, the initial position of the impurity field for the Cauchy problem coincides with the final position of the impurity field at the last time step of solving the direct problem (Fig. 2):

$$N_k^{(2)}(t = 0, \vec{X}) = N_k(t = t_{end}, \vec{X}).$$

The superscript 2 at N indicates the fact that these are the values of the initial impurity concentration field for the inverse problem, without an index – for the direct problem. The time step was $dt = -10^{-5}$.

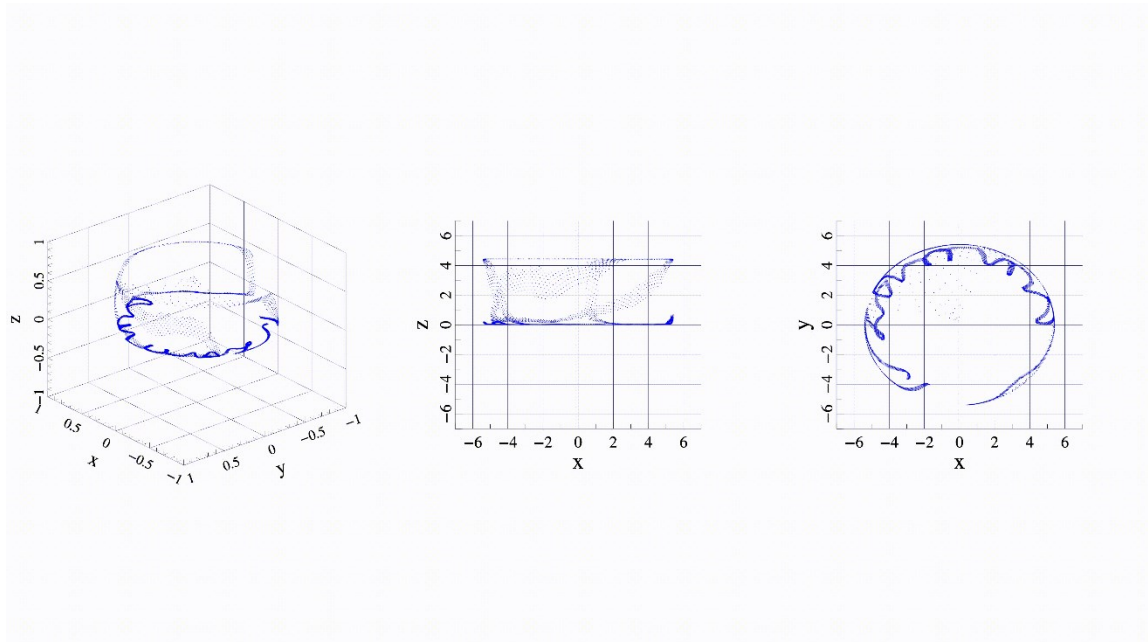


Fig. 3. The result of solving the inverse problem

Fig. 1 and Fig. 3 coincide qualitatively at the final moment of time, which indicates the high accuracy of the resulting solution. The quantitative indicators of the error measure are as follows:

the maximum and average error for each of the coordinates have values:

$$x_{1_max} = 0.0919, x_{1_avg} = 0.045;$$

$x_{2_max} = 0.0921, x_{2_avg} = 0.049;$
 $x_{3_max} = 0.00117, x_{3_avg} = 0.0004.$

At the same time, the distance between the points formed initially (Fig. 1) and the points obtained by solving the inverse problem (Fig. 3) was estimated. The maximum distance is 0.0923, the average value is 0.074.

With respect to the size of the figure, the calculation error is 0.9 and is calculated as follows:

$$E_{otn} = \frac{\max error}{2 * Sq_0} * 100 = \frac{0.0923}{2 * 5} * 100 = 0.923$$

Considering the movement of impurities in this field, it can be assumed that it describes the movement of updrafts, and perhaps the geoglyphs (Fig. 4) on the Nazca plateau were created by the ancient Nazcans based on this effect.



Fig. 4. Example of geoglyphs on the Nazca plateau

Fig. 1, which was used to determine the initial conditions of the Cauchy problem, was formed on the basis of Fig. 4.

Consider the system (1), taking into account the effect of gravity on an impurity dissolved in a liquid. To strengthen the assumption, we will conduct another computational experiment, for this we modify the field (4) by adding a coefficient responsible for the gravitational component, as follows:

$$U(\vec{X}) = \frac{J'_0\left(\frac{r}{\sqrt{2}}\right)}{r\sqrt{2}} \begin{bmatrix} x_2 \sin\left(\frac{x_3}{\sqrt{2}}\right) + \frac{x_1}{\sqrt{2}} \cos\left(\frac{x_3}{\sqrt{2}}\right) \\ -x_1 \sin\left(\frac{x_3}{\sqrt{2}}\right) + \frac{x_2}{\sqrt{2}} \cos\left(\frac{x_3}{\sqrt{2}}\right) \\ 0 \end{bmatrix} + \frac{1}{2} J_0\left(\frac{r}{\sqrt{2}}\right) \begin{bmatrix} 0 \\ 0 \\ \sin\left(\frac{x_3}{\sqrt{2}}\right) \end{bmatrix} - G\vec{k}.$$

In this case, the particles lifted by the field will settle in the $z = 0$ plane over time, and therefore a stop criterion was added for the algorithm, according to which the calculation was stopped if the point along the z axis begins to take negative values at the next iteration. After that, for all subsequent time steps, the value for this point was equal to the last correct value.

During this computational experiment, the following values were changed:

$$t_{start} = 0, t_{end} = 600, dt = 0.001, dt_{save} = 4$$

The value of the parameter $G = 0.0005$.

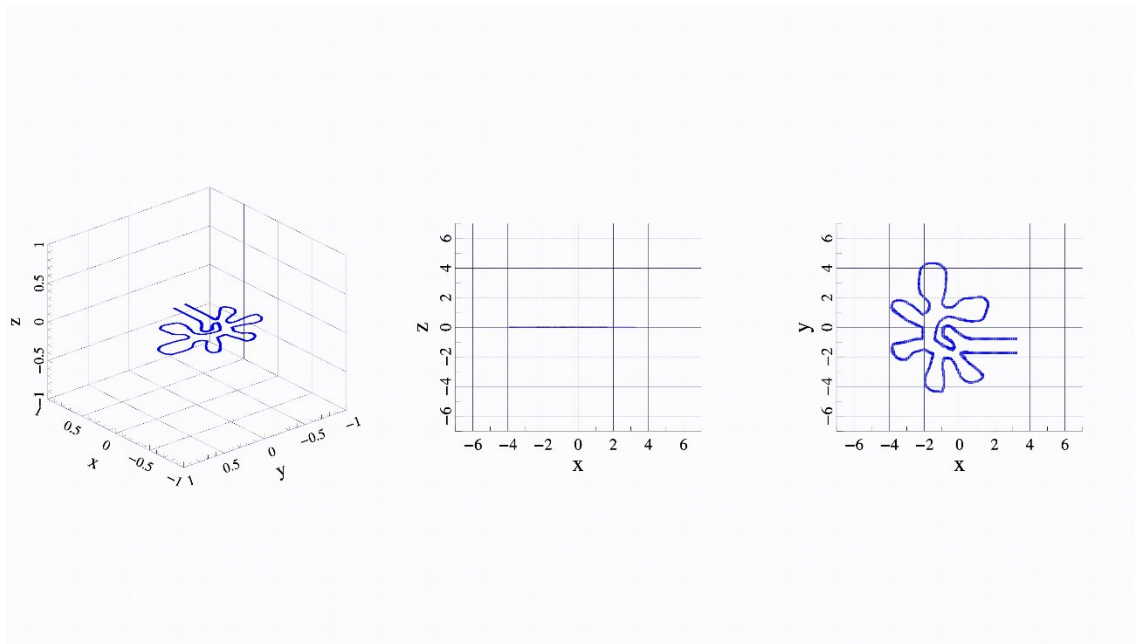


Fig. 5. Sedimentation modeling

As can be seen from the results of mathematical modeling, Fig. 5, by the final moment of time $t = 600$, all particles settled in the plane $z = 0$.

As can be seen from the results of the computational experiment, precipitation fell on the "surface of the earth" in the form of a certain geoglyph, in the future it is possible to develop an algorithm for selecting initial conditions, so that during modeling in the $z = 0$ plane the required image is obtained. To do this, it is necessary to form the initial points in the plane $z = 0.1$, which will occupy the entire area of the base of the cylinder. Then modeling is carried out and, after the impurity falls out, it is necessary to select only those points that form the required image. Then conduct another computational experiment, for which the starting points will be only those that form the image.

In addition, if you look at the zOx projection (fig. 5, average), you can see that the particles in the air also form some arbitrary patterns. With the use of a similar algorithm, only the selection of points will occur not relative to the image obtained in the projection $z = 0$, but relative to the image in the projection zOx . As a result, you can get a tool for forming various aerial illusions / projections in the air.

The work was performed within the framework of the state task of the Federal State University of the Federal Research Center of the Russian Academy of Sciences on the topic FNEF-2024-0001 "Creation and implementation of trusted artificial intelligence systems based on new mathematical and algorithmic methods, models of fast computing implemented on domestic computing systems" (1023032100070-3-1.2.1).

References

1. V. A. Galkin, A.O. Dubovik, D.A. Morgun. Visualization of Flow of a Viscous Incompressible Fluid Corresponding to Exact Solutions of the Navier-Stokes Equations (2024). Scientific Visualization 16.1: 52 - 63, DOI: 10.26583/sv.16.1.05
2. Voloshchuk V.M., Sedunov Yu.S., Coagulation processes in dispersed systems. Leningrad: Gidroieteroizdat. 1975. 320 p.
3. Syromyatnikov P. V., Krivosheeva M.A., Lapina O.N., Nesterenko A. G., Nikitin Yu.G. Modeling of nonstationary diffusion-convection-reaction processes in a multilayer half-space and coupled half-spaces // Ecological Bulletin of the Scientific centers of the BSEC. 2020. Vol. 17. No.1. Part 1. C. 30-41.
4. Vabishevich P. N. Monotonic schemes for convection-diffusion problems with convective transfer in various forms // ZHVM and MF. 2021. Vol. 61. No. 1. pp. 95-107.

5. Lapina O. N., Nesterenko A. G., Nikitin Yu.G., Pavlova A.V. // Modeling of the diffusion – convection process of a contaminant from a periodic source. Ecological Bulletin of the scientific centers of the Black Sea Economic Cooperation. 2022. Vol. 19. No. 1. pp. 25-34.
6. Saritskaia Zh.Yu., Brizitskii R. V. Boundary control problems for nonlinear reaction-diffusion-convection model // Far Eastern Mathematical Journal. 2023. V. 23. No 1. P. 106–111.
7. Aldallo M., Ponikarov S. I., Ponikarov A. S. Modeling of turbulent dynamics and phase transitions in the flow of a mixture of two components in a circular pipe // Bulletin of the Technological University. 2024. Vol.27, No.1
8. V.A. Galkin, A.O. Dubovik On a class of exact solutions of the system of Navier-Stokes equations for an incompressible fluid // Mathematical modeling. 2023. Vol. 35, No. 8. pp. 3-13.
9. Bakhvalov N. S., Zhidkov N. P., Kobelkov G. M. Numerical methods. — M.: Laboratory of Basic Knowledge, 2001. — 630 p. — ISBN 5-93208-043-4. — pp. 363-375.
10. Kalitkin N. N. Numerical methods. St. Petersburg: BHV-Petersburg; 2011. 592 p.
11. MathGL. Access mode: <https://mathgl.sourceforge.net/>.
12. OpenMP. Access mode: <https://web.archive.org/web/20080720100837/http://www.openmp.org/>