60
EKSTU
1958
2018
1958

ВЕСТНИК
Восточно-Казахстанского государственного технического университета им. Д. Серикбаева

ВЫЧИСЛИТЕЛЬНЫЕ ТЕХНОЛОГИИ
Институт вычислительных технологий Сибирского отделения РАН

ИВТ

ISSN 1560-7534
№ 3

2018 № 3

Вычислительные **Т**ехнологии

Вычислительные технологии

НАУЧНЫЙ ЖУРНАЛ ▪ ҒЫЛЫМИ ЖУРНАЛ

Главный редактор
академик Ю.И. Шокин

ИВТ
ИНСТИТУТ ВЫЧИСЛИТЕЛЬНЫХ ТЕХНОЛОГИЙ
Сибирского отделения Российской ака...

EKSTU
1958

ISSN 1561-4212
сентябрь, 2018 г.

Д. Серікбаев атындағы Шығыс Қазақстан мемлекеттік техникалық университетінің

ХАБАРШЫСЫ

ВЕСТНИК
Восточно-Казахстанского государственного технического университета имени Д. Серикбаева

3

СибГУТИ

СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ
И ИНФОРМАТИКИ

Novosibirsk State
Technical University
NSTU

H L R | S
High-Performance Computing Center | Stuttgart

ИВТ

КазИТУ

N* Новосибирский
государственный
университет
*НАСТОЯЩАЯ НАУКА

60
EKSTU
2018
1958

University of Pristina
Kosovska Mitrovica

МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ
«Вычислительные и информационные технологии в науке, технике и образовании»

25-28 сентября

Республика Казахстан, г. Усть-Каменогорск, ул. Серикбаева, 19, 2018 г.

ВЕСТНИК

**Восточно-Казахстанского государственного
технического университета
им. Д. Серикбаева**

Главный редактор

доктор биологических наук, профессор

Ж.К. Шаймарданов

❖

ВЫЧИСЛИТЕЛЬНЫЕ ТЕХНОЛОГИИ

**Институт вычислительных технологий
Сибирского отделения РАН**

Главный редактор

академик

Ю.И. Шокин

———◆———

по материалам Международной конференции
«Вычислительные и информационные технологии
в науке, технике и образовании»
(CITech-2018)
25-28 сентября 2018 года

Том I. Часть II

Усть-Каменогорск-Новосибирск, 2018

Научный комитет
Международной конференции
«Вычислительные и информационные технологии в науке, технике и образовании 2018»

Сопредседатель – Юрий Шокин, действительный член РАН, Институт вычислительных технологий СО РАН, Россия

Сопредседатель – Жасулан Шаймарданов, профессор, Восточно-Казахстанский государственный технический университет им. Д. Серикбаева, Казахстан

Сопредседатель – Бакытжан Жумагулов, действительный член НАН РК, Национальная инженерная академия РК, Казахстан

Максат Калимолдаев, действительный член НАН РК, Институт информационных и вычислительных технологий, Казахстан

Галимкаир Мутанов, действительный член НАН РК, Казахский национальный университет им. аль-Фараби, Казахстан

Нурлан Темирбеков, действительный член НИА РК, Казахстанский инженерно-технологический университет, Казахстан

Аманбек Жайнаков, действительный член НАН КР, Кыргызский государственный технический университет им. И. Раззакова, Кыргызстан

Игорь Бычков, действительный член РАН, Институт динамики систем и теории управления СО РАН, Россия

Виктор Сойфер, действительный член РАН, Самарский государственный аэрокосмический университет им. С.П. Королева, Россия

Александр Стемповский, член-корреспондент РАН, Институт проблем проектирования в микроэлектронике РАН, Россия

Градимир Милованович, действительный член САНУ, Математический институт САНУ, Сербия

Уалихан Абдибеков, член-корреспондент НИА РК, Н.А. Ясавский международный казахско-турецкий университет, Казахстан

Михаил Федорук, член-корреспондент РАН, Новосибирский государственный университет, Россия

Анатолий Федотов, член-корреспондент РАН, Институт вычислительных технологий СО РАН, Россия

Сергей Кабанихин, член-корреспондент РАН, Институт вычислительной математики и математической геофизики СО РАН, Россия

Владимир Шайдуров, член-корреспондент РАН, Институт вычислительного моделирования СО РАН, Россия

Сергей Смагин, член-корреспондент РАН, Компьютерный центр ДВО РАН, Россия

Игорь Бессмертный, профессор, Университет ITMO, Россия

Вальдемар Войчик, профессор, Люблинский технический университет, Польша

Самир Рустамов, доцент, Институт систем управления НАНА, Азербайджан

Павел Венгерик, Люблинский технический университет, Польша

Юрий Крак, профессор, Киевский национальный университет имени Тараса Шевченко, Украина

Януш Партыка, профессор, Люблинский технический университет, Польша

Георг Георг, профессор, Университет Обуда, Венгрия

Бо Эйнарссон, профессор, Университет Линчёпинг, Швеция

Сергей Баутин, профессор, Уральский государственный университет путей сообщения, Россия

Эгон Краузе, профессор, Рейнско-Вестфальский технический университет Ахена, Германия

Сергей Черный, профессор, Институт вычислительных технологий СО РАН, Россия

Пейман Гиви, профессор, Университет Питтсбурга, США

Вагди Жорж Хабаши, профессор, Университет Макгилла, Канада

Андреас Гриеванк, профессор, Берлиинский университет имени Гумбольдта, Германия

Маттиас Мейнке, профессор, Рейнско-Вестфальский технический университет Ахена, Германия

Хранислав Милошевич, профессор, Университет Приштины, Сербия

Владимир Москвичев, профессор, Специальное конструкторско-технологическое бюро «Наука» ИВТ СО РАН, Россия

Вадим Потапов, профессор, Институт вычислительных технологий СО РАН, Россия

Олег Потатуркин, профессор, Институт автоматики и электрометрии СО РАН, Россия

Майкл Реш, профессор, Высокопроизводительный вычислительный центр в Штутгарте, Германия

Карл Рознер, профессор, Технологический университет Дармштадта, Германия

Борис Рябко, профессор, Институт вычислительных технологий СО РАН, Россия

Владимир Садовский, профессор, Институт вычислительного моделирования СО РАН, Россия

Вольфганг Шредер, профессор, Рейнско-Вестфальский технический университет Ахена, Германия

Сергей Турицын, профессор, Университет Астон, Великобритания

Рен-Хун Вонг, профессор, Даляньский технологический университет, Китай

Зиявиддин Юлдашев, профессор, Национальный университет Узбекистана им. Мирзы Улугбека, Узбекистан

Юрий Захаров, профессор, Кемеровский государственный университет, Россия

Дархан Ахмед-Заки, Казахский национальный университет им. аль-Фараби, Казахстан

Томас Бёниш, Высокопроизводительный вычислительный центр в Штутгарте, Германия

Денис Есипов, Институт вычислительных технологий СО РАН, Россия

Денис Дутых, Университет Савойи, Франция

Нина Шокина, Университет Фрайбурга, Германия

Андрей Юрченко, Институт вычислительных технологий СО РАН, Россия

Олег Жижимов, Институт вычислительных технологий СО РАН, Россия

Организационный комитет
Международной конференции
«Вычислительные и информационные технологии
в науке, технике и образовании 2018»

Денис Есипов, Институт вычислительных технологий СО РАН, Россия - секретарь

Алексей Редюк, Институт вычислительных технологий СО РАН, Россия

Сергей Рылов, Институт вычислительных технологий СО РАН, Россия

Олег Сидельников, Институт вычислительных технологий СО РАН, Россия

Олег Гавриленко, ВКГТУ им. Д. Серикбаева, Казахстан

Наталья Денисова, ВКГТУ им. Д. Серикбаева, Казахстан

Назгуль Ердыбаева, ВКГТУ им. Д. Серикбаева, Казахстан

Large eddy simulation the evolution of the cloud explosion of a launch vehicle

S. Abdibekov, D. Zhakebayev, K. Karzhaubayev, and U. Abdibekov

Kazakh National University named after al-Farabi,
71, al-Farabi ave., 050012, Almaty, Kazakhstan
abdibekov_sultan@mail.ru,dauren.zhakebayev@gmail.com,uali1@mail.ru,
kairzhan.k@gmail.com

Abstract In this paper, the evolution of an explosion cloud under the influence of buoyancy force is considered, taking into account turbulent mixing and adiabatic expansion. Numerical simulation on the basis of the solution of the three-dimensional filtered non-stationary Navier-Stokes equation, the continuity equation, the concentration equation, the enthalpy equation, the equation of state for compressible media is carried out. The modified solver is based on the OpenFOAM mathematical physics library. To close the basic equations a viscous model of turbulence is used. The dependence of the cloud lifting height of the launch vehicle explosion on the thermodynamic parameters is obtained based on the large eddy simulation.

Introduction

Emergency fall launch vehicles are often accompanied by ground blast, which yields a gas-dust cloud. The impossibility of carrying out experimental measurements of the mechanical and chemical composition of the cloud, its temperature, interaction with the environment, in connection with the transience of the process of formation and propagation of the cloud, necessitate the use of methods of mathematical modeling. The study of the process of raising an explosion cloud under the influence of Archimedes' force, taking into account turbulent mixing and adiabatic expansion, despite the large number of publications on this topic, remains an urgent task for researchers.

For the first time, the problem of lifting the cloud of an atomic explosion was considered in [1], [2] by Satton and Machta, where it is assumed that the rise of the cloud is caused by Archimedes' force, and the cooling occurs as a result of adiabatic expansion and mixing with the surrounding air. Circulation in the cloud and the influence of inertia are not taken into account, as well as an analytical method for determining the values of the maximum lift height of the cloud by launch vehicle explosion .

In [3], [4], the process of propagation of a turbulent gas jet in a medium of a different density and the initial section of a turbulent jet of a compressible gas were investigated,where on the surface of the explosion cloud turbulent mixing of heated air inside the ring occurs and of cold ambient air.

The problem of the motion of a vortex ring formed after an explosion under the action of gravity was theoretically investigated in [5], where the case of a small density difference inside and outside the cloud was considered, as well as the case when the density difference is large.The volume of hot air formed after an explosion from a shock wave, clouds of the launch vehicle explosion, after increasing to a certain size, tends upward, forming a vortex ring. In the case when the cloud is formed at the surface of the Earth, its rise is accompanied by the raising of a dust column giving a mushroom shape. The main acting force for raising the cloud is Archimedes' power, which arises at different density of air, both inside the cloud and outside the cloud, in the atmosphere. But, in addition to the strength of Archimedes, because of the existence of a vortex ring of circulation, Zhukovsky's force acts on the cloud, directed perpendicularly to the direction of the velocity of movement of the element of the ring. The horizontal component of this force stretches the vortex ring to the sides, and the vertical component - somewhat inhibits the lifting of the ring. The air temperature at the beginning of the rise is large, then falls due to adiabatic expansion. As a result of the study, the dependence of the vertical and horizontal components of the velocity of motion on time, as well as the geometric dimensions of the cloud depending on the power of the explosion were obtained.

The study of the evolution of large-scale thermals in the inhomogeneous atmosphere, produced by powerful explosions, within the framework of the Boussinesq approximation made in the work in 1987 [6].

In the above mentioned works [5], [6] Boussinesq approximation was used, in which the density is considered constant in all terms of the equation of motion, where the density is proportional to the temperature gradient. However, Boussinesq approximation is applicable only for description of flows with relatively small temperature changes and small values of hydrostatic compressibility parameter [7].

With the advent of relatively affordable computing devices, works appeared in which more complex models [7], [8], [9] based on the Navier-Stokes equations for compressible gases were used to describe the process of lifting the explosive cloud. Physically, the use of compressible formulation in equations leads to more adequate results, since density is not always a linear function of the temperature gradient.

In this paper, we consider the raising of the launch vehicle explosion cloud under the influence of Archimedes ' force, taking into account turbulent mixing and adiabatic expansion. The air temperature at the initial moment of the vortex ring is very high, over time it falls due to adiabatic expansion, i.e. reducing the pressure with height and due to turbulent mixing of heated and cold air. Under the influence of Archimedes force, clouds of explosion of heated air will rise into the atmosphere until the temperature is equalized, the density of the gas components of the external and internal heated and cold air due to convective and diffusive mixing. The cooling temperature due to thermal radiation can be neglected, because the length of the radiation path is much larger than the cloud.

Numerical modeling of cloud formation is based on the solution of three-dimensional filtered unsteady Navier-Stokes equation, continuity equation, concentration equation, enthalpy equation of state for compressible media. The viscosity turbulence model is used to close the main equations. The main problem in this paper is the correct description of turbulent transport processes. The explosion power is calculated from the size of the funnel.

The results of numerical modeling of the cloud formation formed during the ground explosion of the Proton-M launch vehicle on July 2, 2013 in the position area of the Baikonur cosmodrome are presented. The dependence of the cloud lifting height in the atmosphere on thermodynamic parameters is shown. The obtained results of the simulation that determines the height of rise of the cloud corresponds to the results of the work of [5], under a low power of the explosion.

Problem statement

There is a fiery half sphere at the initial moment of time at the earth surface, denoted by field G, radius R, initial temperature of ball T_0, and initial three-dimensional density of dry air's gas phases ρ_d, and humid air rho_w, ambient temperature T_1 (Fig. 2).
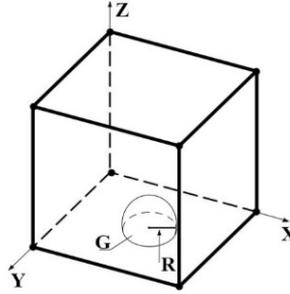
Figure 1. Schematic illustration of the problem statement.

The following filtered equations are used to solve the problem under consideration: The equation of continuity:

$$\frac{\partial \rho_m}{\partial t} + v(\rho_m u_m) = 0 \qquad (1)$$

where u_m is velocity of the gas phase mixture, $\rho_m = \rho_w + \rho_d, u_m = \frac{\nu_m}{\rho_m}$ is air density, $v_m = \frac{\nu_m}{\rho_m}$ is dynamic viscosity, $\nu_m = \nu_* \left(\frac{T}{T_*}\right)^w$, where ν_* is the value of dynamic viscosity at temperature T_*, $\rho_d = \sum_{\alpha=1}^{N_\alpha} S_\alpha \rho_\alpha$; $\sum_{\alpha=1}^{N_\alpha} S_\alpha = 1$ is three-dimensional density of dry airs gas phase, $\rho_w = \sum_{\beta=1}^{N_\beta} S_\beta \rho_\beta$; $\sum_{\beta=1}^{N_\beta} = 1$ three-dimensional density of the water vapors gas phase, $\rho_\alpha = \rho_{*\alpha}(1 - \gamma(T - T_*))$,

$\alpha = 1, \ldots, N_\alpha$; $\rho_\beta = \rho_{*\beta}(1 - \gamma(T - T_*))$, $\beta = 1, \ldots, N_\beta$, also $\rho_{*\alpha}, \rho_{*\beta}$ are density of dry air's gas components and moist air, respectively, at $T_* = 20^0 C$.

The concentration equation:

$$\frac{\partial(\rho_m S_\alpha)}{\partial t} = \nabla(\rho, u_m S_\alpha) = 0, \alpha = 1, \ldots, N_\alpha$$
$$\frac{\partial(\rho_m S_\beta)}{\partial t} = \nabla(\rho, u_m S_\beta) = 0, \beta = 1, \ldots, N_\beta \tag{2}$$

The equation of motion:

$$\frac{\partial(\rho_m u_m)}{\partial t} + \nabla(\rho_m u_m \otimes u_m + \rho_m T_r) = -\nabla p + \nabla T + \rho_m g \tag{3}$$

where g- acceleration occurs under the gravity action, T-– stress tensor of gas phase, t -– time, p -– pressure.

The enthalpy equation:

$$\frac{\partial(\rho_m h_m)}{\partial t} + \nabla(\rho_m h_m u_m) + \nabla(q - T \cdot u_g) =$$
$$= \frac{\partial \rho}{\partial t} - \frac{\partial(\rho_m K_M)}{\partial t} - \nabla(\rho_m K_m u_m) + \rho_m(g u_m) \tag{4}$$

where h_m - enthalpy of gas mixture, heat flux in the gas phase $- -q = -k_g \Delta T$, here k_i - i-th component's conductivity, T- temperature, $K_m = \frac{1}{2}[u, m]^2$- kinetic energy per unit mass of the gas phase.

The temperature equation:

$$T = \frac{h_m}{\frac{1}{\rho_m} \sum_{i=1}^I \rho_i C_i + \frac{1}{\rho_m} \sum_{i=1}^I \rho_i R_i} \tag{5}$$

where C_i - specific heat of gas phase at constant.

The pressure equation: The equation of ideal gas' state is:

$$P = \frac{R_*}{M_{rd}} \rho_d T + \frac{R_*}{M_{rw}} \rho_w T = R_* T \left(\frac{\rho_d}{M_{rd}} + \frac{\rho_w}{M_{rw}} \right) \tag{6}$$

where $R_* = 8.3144598, M_{rd} = \sum_\alpha^{N_\alpha} S_\alpha M_{r\alpha}, M_{rw} = \sum_\beta^{N_\beta} S_\beta M_{r\beta}$.

Initial conditions:

$$u_i(x_1, x_2, x_3, t = 0) = u_0(x_1, x_2, x_3), (x_1, x_2, x_3) \in G,$$

$$u_i(x_1, x_2, x_3, t = 0) = 0, (x_1, x_2, x_3) \notin G,$$

$$T(x_1, x_2, x_3, t = 0) = T_1, (x_1, x_2, x_3) \in G,$$

$$T(x_1, x_2, x_3, t = 0) = T_0, (x_1, x_2, x_3) \notin G,$$

$$S_\alpha(x_1, x_2, x_3, t = 0) = \frac{\rho_\alpha}{\rho_m}, \alpha = 1, \ldots, N_\alpha,$$

$$S_\beta(x_1, x_2, x_3, t = 0) = \frac{\rho_\beta}{\rho_m}, \beta = 1, \ldots, N_\beta.$$

Boundary conditions:

$$\frac{\partial u_i}{\partial n} = 0, \frac{\partial S}{\partial n} = 0, \frac{\partial S_\beta}{\partial n} = 0, \frac{\partial T}{\partial n} = 0, i = 1, 2, 3; \alpha = 1, \ldots, N_\alpha; \beta = 1, \ldots, N_\beta.$$

The algorithm for determining the initial value of fireball's temperature

The equation of total energy consists specific internal energy and kinetic energy. Suppose that in this problem the kinetic energy is zero.

$$E = U \tag{7}$$

where $E = 0.25 \cdot q \cdot t_{ex}$- explosion energy; t_{ex}- explosion time; q - explosion power. One-third of the energy released during the explosion is emitted in radiation form [3]. As a result, the energy enclosed in the fireball at the rise beginning is approximately one quarter of total explosion energy. Specific internal energy in the adiabatic process is expressed by:

$$U = C_v(T_1 - T_0) \tag{8}$$

where C_v is heat capacity of gas in processes with a constant volume, T_1 is ambient temperature for different values of fireball's initial temperature T_0. Substituting equation (8) into equation (7), there is obtained the initial value of the fireball's temperature:

$$T_0 = \frac{E}{C_v} + T_1 \tag{9}$$

Large eddy simulation

Applying the filter to the basic equations (1) - (6), the following equations are obtained [10], [11]:
    The equation of continuity:

$$\frac{\partial \overline{\rho}_m}{\partial t} + \nabla(\overline{\rho}_m \cdot \overline{u}_m) = 0 \tag{10}$$

    The concentration equation:

$$\begin{aligned}
\frac{\partial(\rho_m S_\alpha)}{\partial t} + \nabla(\rho, u_m S_\alpha) = -\nabla G_\alpha, \alpha = 1, \ldots, N_\alpha \\
\frac{\partial(\rho_m S_\beta)}{\partial t} + \nabla(\rho, u_m S_\beta) = -\nabla G_\beta, \beta = 1, \ldots, N_\beta,
\end{aligned} \tag{11}$$

where $G_\alpha = \overline{\rho}_m(\overline{S_\alpha u_m} \overline{S_\alpha u_m}) = -\frac{\mu_t}{Pr_t}\Delta\overline{S_\alpha}$, $G_\beta = \overline{\rho}_m(\overline{S_\beta u_m} \overline{S_\beta u_m}) = -\frac{\mu_t}{Pr_t}\Delta\overline{S_\beta}$ describe the contribution of the sub-grid turbulent scales for gas components concentration equation.

The equation of motion:

$$\frac{\partial(\rho_m u_m)}{\partial t} + \nabla(\rho_m u_m \otimes u_m + \rho_m T_r) = -\nabla p + \nabla \cdot T + \rho_m g - \nabla \cdot B \qquad (12)$$

where $B = \overline{\rho_m} = (\overline{u_m \otimes u_m} - \overline{u_m} \otimes \overline{u_m} = \frac{2}{d}K_t I - 2\mu_t S_m$ - subgrid tensor responsible for small-scale structures, that need to be modeled.

The enthalpy equation:

$$\frac{\partial(\rho_m h_m)}{\partial t} + \nabla(\rho_m h_m u_m) + \nabla(q - T \cdot u_g) =$$
$$= \frac{\partial\rho}{\partial t} - \frac{\partial(\rho_m K_M)}{\partial t} - \nabla(\rho_m K_m u_m) + \rho_m(g u_m) - \nabla(Q + Q_k) \qquad (13)$$

where $Q = \overline{\rho}_m(\overline{h_m u_m} - \overline{h}_m\overline{u}_m) = -\frac{\mu_t}{Pr_t}\Delta\overline{h}_m$, $Q_K = \overline{\rho}_m(\overline{K_m u_m} - \overline{K}_m\overline{u}_m) = -\frac{\mu_t}{Pr_t}\Delta\overline{K}_m$, describe the contribution of subgrid turbulent scales [12], [13] , Pr_t- turbulent Prandtl number, μ_t- turbulent viscosity.

Numerical method

Three dimensional numerical simulation of equation (10) - (13) is performed with indicated initial and boundary conditions to obtain non-stationary fields of unknown variables. The implementation of the numerical algorithm is based on the finite volume method on unstructured grid using the OpenFOAM class library for C++ with an open GPL license. Using C++ templates, the OpenFOAM library allows to quickly create effective solvers and utilities for pre and post processing of modeling results due to the high level of abstraction. Classes and functions in the OpenFOAM library have implicit means for parallelizing computational procedures, due to the numerical calculation on multiprocessor computing systems does not require specific adaptations in the program code. In the finite volume method [14], partial differential equations are integrated over the volume of arbitrary cell, after the Gauss-Ostrogradsky theorem is used to translate volume integrals into surface integrals. It is necessary to interpolate unknown values on each face of finite volume, when calculating flows across finite volume boundaries. Such characteristics as accuracy and stability depend on the interpolation method's choice. Integration over time is carried out using the Crank-Nicholson scheme, the Courant number was maintained at 0.5.

Both for convective and diffusion terms implicit schemes were used to ensure the stability of numerical calculations.

The PISO procedure was used to bind the velocity and pressure fields, as well as to implement the law of conservation of mass [15]. In the equations of motion

and mass conservation are used explicit representations of pressure and gravity fields. Three-dimensional sampling has a second order of accuracy. The PISO algorithm consists of one step of predictor and several steps of proof-readers. An intermediate velocity field is found using the pressure field from the previous time layer in the predictor step. Velocity and pressure fields are corrected to increase the accuracy and reduce the mass defect in the conservation equation at each step of the corrector. The system of linear algebraic equations obtained as a result of the transport equation discretization is solved by the conjugate gradient method with the precedent of Khaletsky for the pressure equation and by the method of bi-conjugate gradients with a preconditioner of incomplete LU factorization.

The algorithm of numerical solution is developed as follows:

1. Solving the conservation equation for mixture density by an explicit method using flows from the previous time layer.

2. Solution of the transport equation for gas components.

3. Solving the momentum conservation equation using the pressure field from the previous iteration.

4. Solving the enthalpy transfer equation using the pressure field from the previous iteration.

5. The solution of a Poisson equation for the pressure.

6. Modification of density and velocity fields based on the new pressure field.

7. Resume iteration from step 5 to reduce the mass defect in the mass conservation equation.

8. The calculation of the subgrid viscosity.

9. Assess the accuracy of the solution and move to step 2 if necessary.

Simulation results

There are results of numerical simulation of cloud formation and dynamics, formed during a surface explosion of a carrier rocket. Numerical simulation of the gas-dust cloud formation stage in the first minutes of the accident was carried out in a cubic area with the physical size of the cube edge 1280 m and the calculated grid $128 \times 128 \times 128$. Ground explosion is accompanied by the funnel formation at Fig.2 Funnels dimensions depend mainly on the explosion power and soil-soil type. The explosion power and depth of the funnel are related by the [6], [7]:

$$q = K_B W^3 (0.4 + 0.6n^3) \tag{14}$$

where q – explosion power; $K_B = 1,35$ - design specific consumption of explosives, kg/m^3; $W = 5$ – depth of the funnel, m; $n = 2$– explosion index. Formula (14) allows to calculate the explosion power at a known depth of the funnel, it turned out q=0,878 t.

The maximum lifting height (km) of the explosion cloud is determined by The setton formula [1]-[5]: $H = 0.665q^{0,276}$.

Table 1. Meteorological parameters for 02.07.2013 (08:00), platform 92 (altitude about 100 m)

Height, m	Air Temperature C	Air Humidity, %	Atmospheric pressure, mmHg	Wind Direction, Deg.	Wind Speed, m/s
10	18.6	45	738.5	10	6-8
20				20	7-8

Table 2. The parameters of the explosion of the launch vehicle

stage number of launch vehicle	1+2+3
The remainder of the propellant at the time of the explosion (UDMH + at), t	0.703
An indicator of the impact of the explosion (n)	2
The power of the explosion, t	0.878
The depth of the crater, m	5
The radius of the crater, m	10
The diameter of the crater, m	20
The volume of the crater, m	785.4
Mass ejected from the funnel of the soil, t	1178.2
Shaft height (parapet) funnel, m	1.3
The radius of the cloud of explosion, m	2.6
The volume of the cloud of explosion, m	74.05
The energy of the explosion, 10^9 J.	3.67
Raising the height of the clouds, m	288

Table 3. Physico-chemical properties of cloud gases

| No.	Name	Molar mass, g/mol	Density at 0^0C, kg/m^3	Kinematic viscosity, m^2/s	Dynamic viscosity, PA at 0^0C	Diameter, m	Fraction, %
1	Carbon monoxide	28.01	1.25	$1329.6 \cdot 10^{-8}$	$1662 \cdot 10^{-8}$	$0.32 \cdot 10^{-9}$	15
2	Carbon dioxide	44.01	1.9768	$693.039 \cdot 10^{-9}$	$1370 \cdot 10^{-8}$	$0.33 \cdot 10^{-9}$	1.95
3	Nitrogen	28.01456	1.251	$2.053 \cdot 10^{-8}$	$1660 \cdot 10^{-8}$	$0.3 \cdot 10^{-9}$	78
4	Nitric oxide	30.00061	1.3402	$1343.6 \cdot 10^{-8}$	$1780 \cdot 10^{-8}$	$0.3 \cdot 10^{-9}$	2.5
5	Nitrogen dioxide	46.0055	2.0527	$829.72 \cdot 10^{-8}$	$1112 \cdot 10^{-8}$	$0.28 \cdot 10^{-9}$	2.5
6	Water vapor	18.01528	998.2	$101.2 \cdot 10^{-8}$	$101000 \cdot 10^{-8}$	$0.29 \cdot 10^{-9}$	0.05

a)   b)

Figure 2. a) Explosion at the emergency fall place (30 seconds); b) formation of the cloud (1 minute after the explosion)

At the first seconds of the accident cloud takes on a mushroom shape, where a vortex ring is observed on the upper part, as can be seen from figure 3. At the initial time, the vortex ring's temperature is large and equal to 1800 K, for 5.5 seconds the temperature drops substantially to 400 K due to adiabatic expansion and turbulent mixing of the cloud's heated air and the environment's cold air. The drop in the cloud's temperature after 5.5 seconds occurs at a lower rate, because at these times the temperature changes due to turbulent mixing. The hot ball of heated air rises to the atmosphere until the temperature, the density of the external and internal heated and cold air's gas components equals, under the influence of buoyancy force. Effect of thermal radiation was not taken into account while performing numerical simulation. Fig.3 shows the dynamics of the concentration in the cloud. Fig.4 shows the change graphs in the height of the cloud rise, maximum temperature in the cloud, volume of the cloud as a time function.

Conclusion

There are obtained results of numerical simulation of cloud formation. There are determined cloud's geometric characteristics raised as a result of surfacing: height of the cloud, the cloud volume, and the shape of the vortex ring in the cloud. Comparison of the cloud rise height as a function of the explosion power with the analytic formula of Satton confirmed the applicability of the mathematical model used to the cloud formation problem in a surface explosion of a carrier rocket. The explosion power is calculated from the funnel size. The results of numerical simulation of cloud formation are obtained. The geometric characteristics of the cloud raised as a result of ascent are determined: the height of the cloud, the volume of the cloud, the shape of the vortex ring in the cloud. The blast power is calculated from the size of the crater. The obtained results of the simulation, which determines the height of the cloud elevation corresponds to the results of Onufriev [5], at low explosion power. In conclusion, we note that the results of this study allow us to estimate the geometric characteristics of raised cloud, the concentration of gas components mixture in the cloud at different instants of
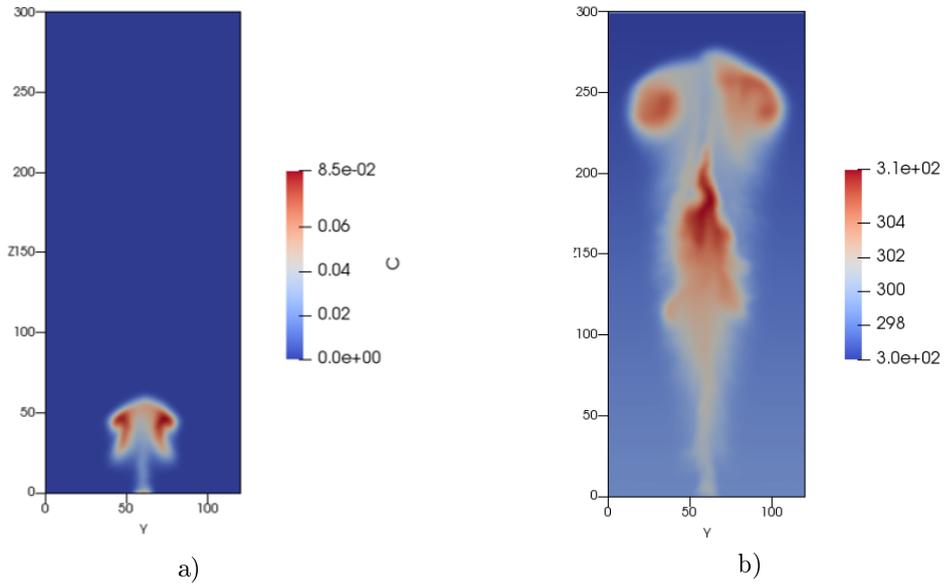
Figure 3. Distribution dynamics of the mixture concentration field at the initial value of explosion energy $E = 3.67 \cdot 10^9$ J in cross section $x = 60m$: a) t = 5 sec; b) t = 35 sec.
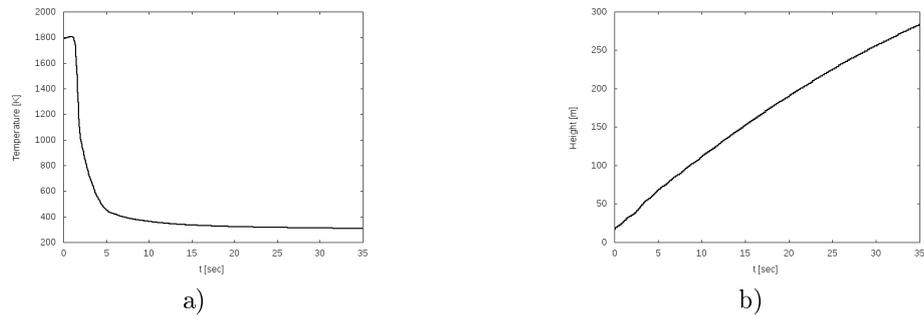
Figure 4. – Change of a) temperature; b) the formed cloud height with the explosion power q=0.878 ton.

time. Such an opportunity is invaluable in absence of experimental data on the cloud formed as a result of an accident. The obtained results allow conducting a primary assessment of the accident impact on the environment.

References

1. Sutton, O.G. : The Atom bomb as an experiment in convection. J.Weather. vol. 2, № 4, pp. 20 (1947)

2. Machta, L.: Eintrainment and the maximum height of an atomic cloud. Bull. Amer. Meteor. Soc. vol. 31, № 6, pp. 215–216 (1950)
3. Vulis, L. A., Terekhina, N. A.: Distribution of turbulent gas jet in a medium of different density.Journal of technical physics. vol. 26, No. 6, (1956)
4. Ershin, S. A., Sakipov Z. B.: Study on the initial phase turbulent jet of compressible gas. Journal of technical physics. vol.2. No. 1 (1959)
5. Onufriev, A. T.: Theory of motion of a vortex ring under the action of gravity. The rise of a cloud of an atomic explosion. J. Appl. No. 2, pp. 3-15 (1967)
6. Gostintsev, Y. A., Solodovnik A. F.: Powerful turbulent thermals in a stably stratified atmosphere. Numerical simulation. J. Appl. No. 1. pp. 47 (1987)
7. Lapin Yu. V., Strelets, M. Kh.: Internal flow of gas mixtures. pp. 368. Moscow: Science (1989)
8. Zatevakhin, M. A.: Turbulent Thermal in a Humid Atmosphere. High Temperature. vol. 39, N 4, pp. 532-539 (2001)
9. Veremey, N. E., Dovgalyuk, Y. A., Stankova E. N.: Numerical simulation of convective clouds developing in the atmosphere in emergency situations. Proceedings of the Russian Academy of Sciences. Physics of atmosphere and oceans. vol. 43, no. 6, pp. 792-806 (2007)
10. Garnier, E., Adams, N., and Sagaut, P.: Large Eddy Simulation for Compressible Flows, doi:10.1007/978-90-481-2819-8, Springer, pp.276. The Netherlands (2009)
11. Moin, P., Squires, K., Cabot, W., and Lee, S.: A dynamic subgrid scale model for compressible turbulence and scalar transport. Phys. Fluids a-Fluid. vol.3, 2746, doi:10.1063/1.858164 (1991)
12. Nicoud, F. and Ducros, F.: Subgrid-scale stress modelling based on the square of the velocity gradient tensor, Flow Turbul. Combust. vol 62, pp. 183-200, doi:10.1023/a:1009995426001 (1999)
13. Piscaglia, F., Montorfano, A., and onorati, A.: Towards the LES simulation of IC engines with parallel topologically changing meshes, SAE Int. J. Engines. vol. 6, pp. 926-940, doi:10.4271/2013-01-1096 (2013)
14. Ferziger and Perić. Computational Methods for Fluid Dynamics. - 3. Rev. ed. - Berlin; Heidelberg; New York; Barcelona; Hong Kong; London; Milan; Paris; Tokyo: Springer, (2002)
15. Issa, R. I., Gosman, A. D., and Watkins, A. P.: The Computation of Compressible and Incompressible Recirculating Flow by a Non-Iterative Implicit Scheme, Journal of Computational Physics. vol. 62, pp 66-82 (1986)

The sweep method for seven point difference equations

A. Abdibekova, D. Zhakebayev, B. Zhumagulov, and A. Assylbekuly

al-Farabi Kazakh National University, Almaty, Kazakhstan,
The National Engineering Academy of the Republic of Kazakhstan
a.aigerim@gmail.com,dauren.zhakebayev@gmail.com

Abstract The results of a numerical method of cyclic sweep for solving seven point difference equations are presented. A numerical algorithm for seven diagonal matrix systems is developed using the cyclic sweep method, taking into account periodic boundary conditions for problems of numerical analysis in spherical, cylindrical coordinate systems. The given algorithm consists of two stages of sweep methods, which yields sixth order accuracy in space. A comparative analysis of the results with the exact solution is made, and it has good matching.The developed algorithm can be applied in solving problems of hydrodynamics in cylindrical, spherical areas.

Keywords: Seven-point difference equations, periodic boundary conditions, finite difference method, cyclic - seven diagonal matrix.

Introduction

In the solutions of problems of hydrodynamics, magnetohydrodynamics and other domains, there are quite large systems of equations with special boundary conditions that take a long time to compute, where sometimes the solution may be unsatisfactory.

In the solution of boundary value problems, linear and nonlinear systems of the equation are found, taking into account periodic boundary conditions. When using three-point approximation and five-point approximation for boundary value problems of ordinary differential equations with constant and variable coefficients, it is possible to obtain, respectively, the second and fourth order of accuracy.

In [1], a periodic solution of three-point difference equations in cylindrical and spherical coordinates was proposed. Such equations arise when ordinary differential equations of the second order are approximated.The appearance of a system of linear equations with a periodic five diagonal matrix is associated with the fourth-order approximation by the method of finite differences of the differential equation taking into account periodic boundary conditions.In [7], the shallow water equations were solved by the finite element method using the two-step method of Numerov-Galerkin, where numerical solutions of cyclic five

- diagonal approximation were required and the fourth order of approximation in space was obtained.

A new method for solving symmetric periodic three-diagonal linear systems was proposed in [3], and later it was modified to adapt to work with matrices having a special structure. In the paper [4] is adapted to the case where the matrix coefficients are strongly diagonally dominant and have a five-point approximation.

Cyclic and cyclic block five diagonal systems of linear equations that occur in the numerical solution of one-dimensional or multidimensional boundary-value problems subject to periodic boundary solutions. These systems can be classified as sparse linear systems [5]. There is a relatively large number of good general and special target programs that can be used to solve these systems through direct or iterative methods [6]-[7].

In this paper, an algorithm for solving cyclic seven-point approximation for numerical analysis problems in spherical, cylindrical coordinate systems is developed. This algorithm consists of two stages of run and allows to obtain an approximation in space to the sixth order of accuracy.

Problem statement

Let it be required to find the solution of the following system of seven point equations:

$$
\begin{cases}
a_1 u_{N-2} + b_1 u_{N-1} + c_1 u_N + d_1 u_1 + e_1 u_2 + f_1 u_3 + g_1 u_4 = z_1, i = 1, \\
a_2 u_{N-1} + b_2 u_N + c_2 u_1 + d_2 u_2 + e_2 u_3 + f_2 u_4 + g_2 u_5 = z_2, i = 2, \\
a_3 u_N + b_3 u_1 + c_3 u_2 + d_3 u_3 + e_3 u_4 + f_3 u_5 + g_3 u_6 = z_3, i = 3, \\
a_i u_{i-3} + b_i u_{i-2} + c_i u_{i-1} + d_i u_i + e_i u_{i+1} + \\
\qquad\qquad\qquad + f_i u_{i+2} + g_i u_{i+3} = z_i, 4 \le i \le N-3, \\
a_{N-2} u_{N-5} + b_{N-2} u_{N-4} + c_{N-2} u_{N-3} + d_{N-2} u_{N-2} + \\
\qquad\qquad + e_{N-2} u_{N-1} + f_{N-2} u_N + g_{N-2} u_1 = z_{N-2}, i = N-2, \\
a_{N-1} u_{N-4} + b_{N-1} u_{N-3} + c_{N-1} u_{N-2} + d_{N-1} u_{N-1} + \\
\qquad\qquad + e_{N-1} u_N + f_{N-1} u_1 + g_{N-1} u_2 = z_{N-1}, i = N-1, \\
a_N u_{N-3} + b_N u_{N-2} + c_N u_{N-1} + d_N u_N + \\
\qquad\qquad\qquad + e_N u_1 + f_N u_2 + g_N u_3 = z_N, i = N.
\end{cases}
\tag{1}
$$

or in vector form

$$ AU = Z $$

where $U = (u_1, u_2, \ldots, u_N)$ - vector of unknowns, $Z = (z_1, z_2, \ldots, z_N)$ –the vector of the right parts, and the matrix is a seven-diagonal square matrix of

dimension $N \times N$.

$$A = \begin{bmatrix} d & e & f & g & 0 & \ldots & \ldots & \ldots & \ldots & 0 & a & b & c \\ c & d & e & f & g & \ddots & & & & \vdots & 0 & a & b \\ b & c & d & e & f & g & \ddots & & & \vdots & \vdots & 0 & a \\ a & b & c & d & e & f & g & \ddots & & \vdots & \vdots & \vdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots & \vdots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & \ldots & \ldots & \ldots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & g \\ g & 0 & \ldots & \ldots & \ldots & \ldots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & f \\ f & g & 0 & \ldots & \ldots & \ldots & \ldots & 0 & \ddots & \ddots & \ddots & \ddots & e \\ e & f & g & 0 & \ldots & \ldots & \ldots & \ldots & 0 & a & b & c & d \end{bmatrix}_{N \times N}$$

Numerical method

For the numerical solution of the system (1) consists of two stages. At the first stage, the method of seven diagonal sweeps is used to find $u_1, u_2, \ldots, u_{N-3}$. At the second stage is looking for u_{N-2}, u_{N-1}, u_N.

$$E = \begin{bmatrix} d & e & f & g & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ c & d & e & f & g & \ddots & & & & \vdots \\ b & c & d & e & f & g & \ddots & & & \vdots \\ a & b & c & d & e & f & g & \ddots & & \\ 0 & a & b & c & d & e & f & g & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & g \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & f \\ \vdots & & & & \ddots & \ddots & \ddots & \ddots & \ddots & e \\ 0 & \ldots & \ldots & \ldots & \ldots & 0 & a & b & c & d \end{bmatrix}_{(N-3) \times (N-3)}$$

At the first stage we find $u_1, u_2, \ldots, u_{N-3}$ by removing from the matrix A the last 3 rows and the last 3 columns we build seven diagonal matrix with size $(N - 3) \times (N - 3)$.

Then the system (1) is reduced to the following form:

$$
\begin{cases}
d_1 u_1 + e_1 u_2 + f_1 u_3 + g_1 u_4 = z_1, i = 1, \\
c_2 u_1 + d_2 u_2 + e_2 u_3 + f_2 u_4 + g_2 u_5 = z_2, i = 2, \\
b_3 u_1 + c_3 u_2 + d_3 u_3 + e_3 u_4 + f_3 u_5 + g_3 u_6 = z_3, i = 3, \\
a_i u_{i-3} + b_i u_{i-2} + c_i u_{i-1} + d_i u_i + e_i u_{i+1} + \\
\qquad\qquad + f_i u_{i+2} + g_i u_{i+3} = z_i, 4 \le i \le M - 3, \\
a_{M-2} u_{M-5} + b_{M-2} u_{M-4} + c_{M-2} u_{M-3} + d_{M-2} u_{M-2} + \\
\qquad\qquad + e_{M-2} u_{M-1} + f_{M-2} u_M = z_{M-2}, i = M - 2, \\
a_{M-1} u_{M-4} + b_{M-1} u_{M-3} + c_{M-1} u_{M-2} + \\
\qquad\qquad + d_{M-1} u_{M-1} + e_{M-1} u_M = z_{M-1}, i = M - 1, \\
a_M u_{M-3} + b_M u_{M-2} + c_M u_{M-1} + d_M u_M = z_M, i = M.
\end{cases}
\tag{2}
$$

where $M = N - 3$.

We proceed to the construction of the algorithm for solving the system (2). For the solution of system (2) sweep method is used for the seven point equations. The solution of the system (2) will be found in the form:

$$
u_i = s_i - p_i u_{i+1} - q_i u_{i+2} - r_i u_{i+3}, \quad i = M - 3, M - 4, \ldots, 1.
\tag{3}
$$

$$
u_{M-2} = s_{M-2} - p_{M-2} u_{M-1} - q_{M-2} u_M, \quad i = M - 2.
\tag{4}
$$

$$
u_{M-1} = s_{M-1} - p_{M-1} u_M, \quad i = M - 1.
\tag{5}
$$

For the implementation of $(3) - (5)$ you need to set U_M, and define sweep method coefficients of s_i, p_i, q_i, r_i . Using equation (3), we express u_{i-1}, u_{i-2}, u_{i-3} through u_i, u_{i+1}, u_{i+2}.

$$
u_{i-1} = s_{i-1} - p_{i-1} u_i - q_{i-1} u_{i+1} - r_{i-1} u_{i+2},
\tag{6}
$$

$$
\begin{aligned}
u_{i-2} &= s_{i-2} - p_{i-2} u_{i-1} - q_{i-2} u_i - r_{i-2} u_{i+1} = \\
&= s_{i-2} - p_{i-2}(s_{i-1} - p_{i-1} u_i - q_{i-1} u_{i+1} - r_{i-1} u_{i+2}) - \\
&\quad - q_{i-2} u_i - r_{i-2} u_{i+1} = (s_{i-2} - p_{i-2} s_{i-1}) + (p_{i-2} p_{i-1} - q_{i-2}) u_i) + \\
&\quad + (p_{i-2} q_{i-1} - r_{i-2}) u_{i+1} + p_{i-2} r_{i-1} u_{i+2}.
\end{aligned}
\tag{7}
$$

$$
\begin{aligned}
u_{i-3} &= s_{i-3} - p_{i-3} u_{i-2} - q_{i-3} u_{i-1} - r_{i-3} u_i = s_{i-3} - \\
&- p_{i-3}[(s_{i-2} - p_{i-2} s i - 1) + (p_{i-2} p_{i-1} - q_{i-2}) + (p_{i-2} q_{i-1} - r_{i-2}) u_{i+1} + \\
&+ p_{i-2} r_{i-1} u_{i+2}] - q_{i-3}(s_{i-1} - p_{i-1} u_i - q_{i-1} u_{i+1} - r_{i-1} u_{i+2}) = \\
&= [s_{i-3} - p_{i-3}(s_{i-2} - p_{i-2} s_{i-1}) - q_{i-3} s_{i-1}] +
\end{aligned}
\tag{8}
$$

$$+[-p_{i-3}(p_{i-2}p_{i-1} - q_{i-2} + q_{i-3}p_{i-1} - r_{i-3})u_i +$$

$$+[-p_{i-3}(p_{i-2}q_{i-1} - r_{i-2}) + q_{i-3}q_{i-1}]u_{i+1} +$$

$$+[-p_{i-3}p_{i-2}r_{i-1} + q_{i-3}r_{i-1}]u_{i+2}.$$

Substituting (6) - (8) into equation sysems (2) for $4 \leq i \leq M - 3$ we obtain

$$[a_i[s_{i-3} - p_{i-3}(s_{i-2} - p_{i-2}s_{i-1}) - q_{i-3}s_{i-1}] + b_i(s_{i-2} - p_{i-2}s_{i-1}) + c_is_{i-1}] +$$

$$+[a_i[-p_{i-3}(p_{i-2}p_{i-1} - q_{i-2}) + q_{i-3}p_{i-1} - r_{i-3}] + b_i(p_{i-2}p_{i-1} - q_{i-2}) -$$

$$-c_ip_{i-1} + d_i]u_i + [a_i[-p_{i-3}(p_{i-2}q_{i-1} - r_{i-2}) + q_{i-3}q_{i-1}] +$$

$$+b_i(p_{i-2}q_{i-1} - r_{i-2}) - c_iq_{i-1} + e_i]u_{i+1} + [a_i[-p_{i-3}p_{i-2}r_{i-1} +$$

$$+q_{i-3}r_{i-1}] + b_ip_{i-2}r_{i-1} - c_ir_{i-1} + f_i]u_{i+2} + g_iu_{i+3} = z_i.$$

or

$$u_i = \frac{1}{\Delta_i}(z_i - [a_i[s_{i-3} - p_{i-3}(s_{i-2} - p_{i-2}s_{i-1}) - q_{i-3}s_{i-1}] +$$

$$+b_i(s_{i-2} - p_{i-2}s_{i-1}) + c_is_{i-1}]) - \frac{1}{\Delta_i}([a_i[-p_{i-3}(p_{i-2}q_{i-1} - r_{i-2}) +$$

$$+q_{i-3}q_{i-1}] + b_i(p_{i-2}q_{i-1} - r_{i-2}) - c_iq_{i-1} + e_i])u_{i+1} - \frac{1}{\Delta_i}g_iu_{i+3} - \qquad (9)$$

$$-\frac{1}{\Delta_i}([b_ip_{i-2}r_{i-1} - c_ir_{i-1} + f_i + a_i[-p_{i-3}p_{i-2}r_{i-1} + q_{i-3}r_{i-1}])u_{i+2},$$

where
$\Delta_i = [a_i[-p_{i-3}(p_{i-2}p_{i-1} - q_{i-2}) + q_{i-3}p_{i-1} - r_{i-3}] + b_i(p_{i-2}p_{i-1} - q_{i-2}) - c_ip_{i-1} + d_i].$

Comparing the expression (9) with (3):

$$s_i = \frac{1}{\Delta_i}(z_i - a_i[s_{i-3} - p_{i-3}(s_{i-2} - p_{i-2}s_{i-1}) - q_{i-3}s_{i-1}] +$$

$$+ b_i(s_{i-2} - p_{i-2}s_{i-1}) + c_is_{i-1}])$$

$$p_i = \frac{1}{\Delta_i}([a_i[-p_{i-3}(p_{i-2}q_{i-1} - r_{i-2}) + q_{i-3}q_{i-1}] +$$

$$+ b_i(p_{i-2}q_{i-1} - r_{i-2}) - c_iq_{i-1} + e_i]) \qquad (10)$$

$$q_o = \frac{1}{\Delta_i}([a_i[-p_{i-3}p_{i-2}r_{i-1} + q_{i-3}r_{i-1}] + b_ip_{i-2}r_{i-1} + f_i])$$

$$r_i = \frac{1}{\Delta_i}g_i.$$

The recurrence relations (10) connects s_i, p_i, q_i, r_i with s_{i-1}, p_{i-1}, q_{i-1}, r_{i-1}, s_{i-2}, p_{i-2}, q_{i-2}, r_{i-2} and $s_{i-3}, p_{i-3}, q_{i-3}, r_{i-3}$. Therefore, if will be set to s_i, p_i, q_i, r_i, then by the formulas (10) can successively find all the coefficients of the sweep method s_i, p_i, q_i, r_i for $3 \leq i \leq M - 1$.

Let's find s_i, p_i, q_i, r_i for $i = 1, 2, 3$. From equation (3) and equation systems (2) for $i = 1$:

$$u_1 = s_1 - p_1 u_2 - q_1 u_3 - r_1 u_4 \qquad (11)$$

$$d_1 u_1 + e_1 u_2 + f_1 u_3 + g_1 u_4 = z_1$$

$$u_1 = \frac{z_1}{d_1} - \frac{e_1}{d_1} u_2 - \frac{f_1}{d_1} u_3 - \frac{g_1}{d_1} u_4 \qquad (12)$$

Comparing the expression (11) with (12) :

$$s_1 = \frac{z_1}{d_1}; \quad p_1 = \frac{e_1}{d_1}; \quad q_1 = \frac{f_1}{d_1}; \quad r_1 = \frac{g_1}{d_1}$$

From equation (3) and equation (2) for $i = 3$:

$$u_2 = s_2 - p_2 u_3 - q_2 u_4 - r_2 u_5 \qquad (13)$$

$$c_2 u_1 + d_2 u_2 + e_2 u_3 + f_2 u_4 + g_2 u_5 = z_2$$

$$u_1 = s_1 - p_1 u_2 - q_1 u_3 - r_1 u_4$$

$$c_2(s_1 - p_1 u_2 - q_1 u_3 - r_1 u_4) + d_2 u_2 + e_2 u_3 + f_2 u_4 + g_2 u_5 = z_2$$

$$(d_2 - c_2 p_1) u_2 = z_2 - c_2 s_1 - (e_2 - c_2 q_1) u_3 - (f_2 - c_2 r_2) u_4 - g_2 u_5$$

$$u_2 = \frac{z_2 - c_2 s_1}{d_2 - c_2 p_1} - \frac{e_2 - c_2 q_1}{d_2 - c_2 p_1} u_3 - \frac{f_2 - c_2 r_1}{d_2 - c_2 p_1} u_4 - \frac{g_2}{d_2 - c_2 p_1} u_5 \qquad (14)$$

Comparing the expression (13) with (14)

$$s_2 = \frac{z_2 - c_2 s_1}{d_2 - c_2 p_1}; \qquad p_2 = \frac{e_2 - c_2 q_1}{d_2 - c_2 p_1};$$

$$q_2 \frac{f_2 - c_2 r_1}{d_2 - c_2 p_1}; \qquad r_2 = \frac{g_2}{d_2 - c_2 p_1}.$$

From (7) и системы (2) we immediately obtain:

$$u_1 = (s_1 - p_1 s_2) + (p_1 p_2 - q_1) u_3 + (p_1 q_2 - r_1) u_4 + p_1 r_2 u_5$$

$$u_2 = s_2 - p_2 u_3 - q_2 u_4 - r_2 u_5$$

$$u_3 = s_3 - p_3 u_4 - q_3 u_5 - r_3 u_6 \qquad (15)$$

$$b_3 u_1 + c_3 u_2 + d_3 u_3 + e_3 u_4 + f_3 u_5 + g_3 u_6 = z_3$$

$$b_3((s_1 - p_1 s_2) + (p_1 p_2 - q_1)u_3 + (p_1 q_2 - r_1)u_4 + p_1 r_2 u_5) +$$

$$+c_3(s_2 - p_2 u_3 - q_2 u_4 - r_2 u_5) + d_3 u_3 + e_3 u_4 + f_3 u_5 + g_3 u_6 = z_3$$

$$(d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2)u_3 = z_3 - (b_3(s_1 - p_1 s_2) + c_3 s_2) -$$

$$-(e_3 + b_3(p_1 q_2 - r_1 - c_3 q_2)u_4 - (f_3 + b_3 p_1 r_2 - c_3 r_2)u_5 - g_3 u_6$$

$$u_3 = \frac{z_3 - (b_3(s_1 - p_1 s_2) + c_3 s_2)}{(d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2)} - \frac{(e_3 + b_3(p_1 q_2 - r_1) - c_3 q_2)}{(d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2)}u_4 - \qquad (16)$$

$$-\frac{(f_3 + b_3 p_1 r_2 - c_3 r_2)}{(d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2)}u_5 - \frac{g_3}{(d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2)}u_6.$$

Comparing the expression (15 with (16)

$$s_3 = \frac{z_3 - (b_3(s_1 - p_1 s_2) + c_3 s_2)}{d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2}; \quad p_3 = \frac{e_3 + b_3(p_1 q_2 - r_1) - c_3 q_2}{d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2};$$

$$q_3 = \frac{f_3 + b_3 p_1 r_2 - c_3 r_2}{d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2}; \qquad r_3 = \frac{g_3}{d_3 + b_3(p_1 p_2 - q_1) - c_3 p_2}.$$

In order to find u_{M-2} and u_{M-3} we substitute equation (3) at $i = M - 1$, $i = M - 2$ и $i = M - 3$

$$u_{M-1} = s_{M-1} - p_{M-1}u_M$$

$$u_{M-2} = s_{M-2} - p_{M-2}u_{M-1} - q_{M-2}u_M$$

$$u_{M-3} = s_{M-3} - p_{M-3}u_{M-2} - q_{M_3}u_{M-1} - r_{M-3}u_M$$

We will express u_{M-2} и u_{M-3} through u_M. In order to express u_{M-2} through u_M substitute u_{M-1} to u_{M-2}. And for the expression u_{M-3} through u_M we substitute u_{M-1} and found u_{M-2} substtitute to u_{M-3}.

$$u_{M-2} = (s_{M-2} - p_{M-2}s_{M-1}) + (p_{M-2}p_{M-1} - q_{M-2})u_{M-1}$$

$$u_{M-3} = [s_{M-3} - p_{M-3}(s_{M-2} - p_{M-2}s_{M-1}) - q_{M-3}s_{M-1}] +$$

$$+[-p_{M-3}(p_{M-2}p_{M-1} - q_{M-2}) + q_{M-3}p_{M-1} - r_{M-3}]u_M$$

$$a_M u_{M-3} + b_M u_{M-2} + c_M u_{M-1} + d_M u_M = z_M \qquad (17)$$

We substitute u_{M-1} and found u_{M-2} and u_{M-3} substitute into equation (17):

$$a_M([s_{M-3} - p_{M-3}(s_{M-2} - p_{M-2}s_{M-1}) - q_{M-3}s_{M-1}])u_M +$$

$$+a_M([-p_{M-3}(p_{M-2}p_{M-1} - q_{M-2}) + q_{M-3}p_{M-1} - r_{M-3}])u_M +$$

$$+b_M((s_{M-2} - p_{M-2}s_{M-1}) + (p_{M-2}p_{M-1} - q_{M-2})u_M)+$$

$$+c_M(s_{M-1} - p_{M-1}u_M) + d_M u_M = z_M$$

$$a_M([-p_{M-3}(p_{M-2}p_{M-1} - q_{M-2}) + q_{M-3}p_{M-1} - r_{M-3}])u_M+$$

$$+a_M(b_M(p_{M-2}p_{M-1} - q_{M-2}) - c_M p_{M-1} + d_M)u_M =$$

$$= z_M - a_M([s_{M-3} - p_{M-3}(s_{M-2} - p_{M-2}s_{M-1}) - q_{M-3}s_{M-1}])-$$

$$-b_M(s_{M-2} - p_{M-2}s_{M-1}) - c_M s_{M-1}$$

$$u_m = \frac{1}{\Delta_M}(z_M - a_M(s_{M-3} - p_{M-3}(s_{M-2} - p_{M-2}s_{M-1}) - q_{M-3}s_{M-1})-$$

$$-b_M(s_{M-2} - p_{M-2}s_{M-1}) - c_M s_{M-1})$$

or

$$u_M = s_M$$

$$u_{M-1} = s_{M-1} - p_{M-1}u_M$$

$$u_{M-2} = s_{M-2} - p_{M-2}u_{M-2} - q_{M-2}u_M$$

$$u_i = s_i - p_i u_{i+1} - q_i u_{i+2} - r_i u_{i+3}$$

At the second stage we find u_{N-2}, u_{N-1}, u_N. $\hat{U}$ vector of unknowns with size $(N-3)$, $\hat{z}$, vector of right-hand sides with size $(N-3)$, $\hat{g}_n$, matrix with size $(N-3) \times 3$, which is constructed from the last 3 rows from the original matrix A, $\hat{f}_n$ is a matrix with size $(N-3) \times 3$, which is constructed from the last 3 columns from the original matrix A.

$$\hat{U} = \begin{bmatrix} u_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ u_{N-3} \end{bmatrix}_{(N-3)} \qquad \hat{g} = \begin{bmatrix} g & f & e \\ 0 & g & f \\ \vdots & 0 & g \\ \vdots & \vdots & 0 \\ 0 & \vdots & \vdots \\ a & 0 & \vdots \\ b & a & 0 \\ c & b & a \end{bmatrix}_{(N-3) \times 3}$$

$$\hat{z} = \begin{bmatrix} z_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ z_{N-3} \end{bmatrix}_{(N-3)} \qquad \hat{f}_n = \begin{bmatrix} a & b & c \\ 0 & a & b \\ \vdots & 0 & a \\ \vdots & \vdots & 0 \\ 0 & \vdots & \vdots \\ g & 0 & \vdots \\ f & g & 0 \\ e & f & g \end{bmatrix}_{(N-3)\times 3}$$

Then the system (2) can be rewritten as

$$E\hat{U} + \hat{f}_n \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \hat{z} \tag{18}$$

$$\hat{g}_n^T \hat{U} + \begin{bmatrix} d & e & f \\ c & d & e \\ b & c & d \end{bmatrix} \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} z_{N-2} \\ z_{N-1} \\ z_N \end{bmatrix}$$

Also we have:

$$\hat{z} = \begin{bmatrix} z_{N-2} \\ z_{N-1} \\ z_N \end{bmatrix} = z \tag{19}$$

From (18) we get

$$\hat{U} = E^{-1}\hat{z} - E^{-1}\hat{f}_n \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} \tag{20}$$

Substituting (20) in (19) then we get:

$$\hat{g}_n^T E^{-1}\hat{z} - \hat{g}_n^T E^{-1}\hat{f}_n \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} + \begin{bmatrix} d & e & f \\ c & d & e \\ b & c & d \end{bmatrix} \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} z_{N-2} \\ z_{N-1} \\ z_N \end{bmatrix}$$

$$\left\{ \begin{bmatrix} d & e & f \\ c & d & e \\ b & c & d \end{bmatrix} - \hat{g}_n^T E^{-1}\hat{f}_n \right\} \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} z_{N-2} \\ z_{N-1} \\ z_N \end{bmatrix} - \hat{g}_n^T E^{-1}\hat{z}$$

In the final analysis, there are three values of the unknown vector

$$\begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \left\{ \begin{bmatrix} d & e & f \\ c & d & e \\ b & c & d \end{bmatrix} - \hat{g}_n^T E^{-1}\hat{f}_n \right\}^{-1} \left\{ \begin{bmatrix} z_{N-2} \\ z_{N-1} \\ z_N \end{bmatrix} - \hat{g}_n^T E^{-1}\hat{z} \right\}$$

Test problem

Consider a linear differential equation of the form:

$$u''(x) - u(x) = (1 - 4\pi^2)\sin(2\pi x), \qquad 0 \le x \le 1 \qquad (21)$$

With periodic boundary conditions:

$$u(1) = u(0), \qquad u(-h) = u(1-h) \qquad (22)$$

where h computational grid step $h = \frac{1}{N}$.

Exact solution of the problem $(21) - (22)$ is written as:

$$u(x) = \sin(2\pi x) \qquad (23)$$

The difference operator for approximating the second derivative of the function u at the point x_i can be represented in the following form:

$$\left.\frac{\mathrm{d}^2 u}{\mathrm{d}x^2}\right|_{x_i} \to \Lambda_{xx} = \frac{2u_{i+3} - 27ui + 2 + 270u_{i+1} - 490u_i}{180h^2} +$$

$$+\frac{270u_{i-1} - 27u_{i-2} + 2u_{i-3}}{180h^2} = \frac{180h^2 \left.u''\right|_i + \frac{12960}{8!}h^8 \left.u^{(8)}\right|_i}{180h^2} = \qquad (24)$$

$$= \left.u''\right|_i + \frac{72}{8!}h^6 \left.u^{(8)}\right|_i = \left.u''\right|_i + O(h^6)$$

Thus, the difference operator (24) approximating the second derivative of the function u at the point x_i, has the sixth order of approximation.

For the numerical solution of the problem (21) - (22) the cyclic seven diagonal sweep method is applied.

$$\frac{1}{180h^2} =$$

$$= \begin{bmatrix}
-490+180h^2 & 270 & -27 & 2 & 0 & \dots & \dots & \dots & 0 & 2 & -27 & 270 \\
272 & \ddots & \ddots & \ddots & \ddots & & & \vdots & 0 & 2 & -27 \\
-27 & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots & \vdots & 0 & 2 \\
2 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & 0 \\
0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots & \vdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & 0 & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & 0 \\
0 & \ddots & \ddots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2 \\
2 & 0 & \ddots & \ddots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & -27 \\
-27 & 2 & 0 & \ddots & \ddots & 0 & \ddots & \ddots & \ddots & \ddots & 270 \\
270 & -27 & 2 & 0 & \ddots & \ddots & 0 & 2 & -27 & 270 & -490+180h^2
\end{bmatrix} .$$

$$
\cdot
\begin{bmatrix}
u_1 \\
u_2 \\
u_3 \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
u_{N-2} \\
u_{N-1} \\
u_N
\end{bmatrix}
=
\begin{bmatrix}
z_1 \\
z_2 \\
z_3 \\
\vdots \\
\vdots \\
\vdots \\
(1 - 4\pi^2)\sin(2\pi ih) \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
z_{N-2} \\
z_{N-1} \\
z_N
\end{bmatrix}
$$

Table 1 compares the results of an exact solution with the numerical solution

Table 1. The data of the exact and numerical solution are presented.

X	Exact solution	Numerical solution	Error
0	0.0000000E+00	-6.3122734E-06	6.3122734E-06
4	0.7071068	0.7070988	7.9274178E-06
8	1.000000	0.9999912	8.8214874E-06
12	0.7071068	0.7070988	7.9274178E-06
16	-8.7422777E-08	-6.0498714E-06	5.9624485E-06
20	-0.7071069	-0.7071112	4.3511391E-06
24	-1.000000	-1.000004	4.1723251E-06
28	-0.7071065	-0.7071114	4.8279762E-06
32	1.7484555E-07	-6.3122734E-06	6.4871188E-06

Thus, a numerical method of cyclic sweep was developed to solve seven point difference equations, where an algorithm for solving the seven diagonal matrix system was constructed by cyclic sweep method taking into account periodic boundary conditions. The obtained results have good agreement with the results of the exact solution. The developed algorithm can find its application in the solution of problems of hydrodynamics, magnetohydrodynamics and electrodynamics in cylindrical, spherical regions.

References

1. Samarskii, L.A., Nikolaev, E.S.: Methods for solving grid equations. Moscow: Nauka, (1978).
2. Navon, I.M. Pent: A Periodic Cyclic Penta-diagonal System Solver. Communications in applied Numerical Methods. vol. 3. pp. 63–69. (1987)
3. Rojo, O.: A new method for solving symmetric circulant tridiagonal systems of linear equations. J.Comput. Math. Appl. vol. 20. pp. 61–67. (1990)
4. El-Sayed, S.M., Ivanov, I.G., Petkov, M.G.: A new modification of the Rojo Method for solving symmetric circulant five-diagonal systems of linear equations. J.Comput. Math. Appl., vol. 35, pp. 35–44. (1998)
5. Minchev, B.V., Ivanov, I.G.: A Method for Solving Hermitian Pentadiagonal Block Circulant Systems of Linear Equations. Lecture Notes in Computer Science. pp. 481-488. Springer. (2004)
6. Karawia, A.A.: A Computational Algorithm for Solving Periodic Penta-Diagonal Linear Systems. J. Appl. Math. and Computation. vol. 174. pp. 613-618. (2006)
7. Nemani, S.S.: A fast algorithm for solving Toeplitz penta-diagonal systems Original. J. Appl. Math. and Computation. vol. 215, pp. 3830-3838. (2010)

Mathematical modelling of turbulence energy by finite-difference method

A. Abdibekova, D. Zhakebayev, and B. Zhumagulov

al-Farabi Kazakh National University, Almaty, Kazakhstan,
The National Engineering Academy of the Republic of Kazakhstan
`a.aigerim@gmail.com,dauren.zhakebayev@gmail.com`

Abstract This work deals with the modelling of the decay of turbulent kinetic energy by combining two different numerical methods: finite-difference and spectral methods. The numerical algorithm solves non-stationary three-dimensional Navier-Stokes equations. The hybrid method solves the Navier-Stokes equations by a finite-difference method in combination with cyclic penta - diagonal matrix, which yields fourth-order accuracy in space and second - order accuracy in time. The pressure Poisson equation is solved by the spectral method, which is proposed to speed up the solution procedure. For validation of the developed algorithm the classical problem of the 3-D Taylor and Green vortex flow is considered, and the simulated time-dependent turbulence characteristics of this flow were found to be in excellent agreement with the corresponding analytical solution valid for short times. We also demonstrate that the developed efficient numerical algorithm can be used to simulate the turbulence decay at different flow Reynolds numbers.

Keywords: Taylor and Green vortex problem, finite difference method, Navier-Stockes equation, spectral method, cyclic -penta diagonal matrix

The study of cascade vortex flow is an urgent task, since the study of vortex dynamics leads to correct models and correct understanding of turbulent flows. In this paper we study the evolution of classical vortex problem proposed by Taylor and Green [1] who considered a possibility of solving the Navier-Stokes equations analytically by a method for successive approximations, in order to describe three-dimensional turbulence evolution (specifically energy cascade and viscous dissipation) over time, with the resulting flow now known as the Taylor-Green vortex flow. Their work was motivated by the decay of three-dimensional turbulent flow produced in a wind tunnel, a fundamental process in turbulent flow, due to the grinding down of eddies, produced by nonlinearity of the Navier-Stokes equations. In their work the kinetic energy and its dissipation rate were determined analytically.

Taylor and Green's original analytical investigation is rigorous only for short times. To extend the understanding of the 3D Taylor-Green vortex flow, Brachet *et al* [3] solved the Taylor- Green vortex problem by two methods: numerical solution using the spectral method and power-series analysis in

time. The resulting average kinetic energy and energy spectra at different flow Reynolds numbers were presented and compared. Later, in [3] three dimensional Navier-Stockes equations were numerically integrated with the periodic Taylor-Green initial condition to obtain the flow evolution at Reynolds numbers $Re_\lambda = 140$. In this direct numerical simulation study the slope of energy spectrum was compared with Kolmogorov's $-5/3$ slope in the inertial subrange. Moreover, the compressible Navier-Stokes equations have also been applied to the Taylor-Green vortex problem using large-eddy simulation in [4] at different grid resolutions, and the time evolutions of the kinetic energy and its dissipation rate were compared at different grid resolutions. However, in these reported studies the comparison results of turbulence characteristics, such as dissipation rate and kinetic energy with the exact analytical solution were not reflected .

This work deals with the modelling of turbulent energy decay using a hybrid method combining finite-difference and spectral methods. To simulate the turbulent process the three-dimensional non-stationary Navier-Stokes equation is used. The problem of turbulence decay is solved at different Reynolds numbers: $Re = 100; Re = 300; Re = 600$ by using the hybrid method, where the equation of motion is solved by using a finite-difference method in combination with cyclic penta-diagonal matrix, which ensures a high-order accuracy, and a spectral method for solution of the Poisson equation, which is highly efficient. For validation of the developed algorithm the classical problem of Taylor and Green [1] was re-considered as a model of decaying turbulence.

Analytical solution of the Taylor-Green vortex problem

We duplicate the classical example proposed in [1] in order to validate the numerical simulation of increasing order of accuracy in time and in space $O(dt^2, h^4)$, with efficient acceleration for sequential algorithm. Starting from a simple incompressible three-dimensional initial condition of the form.

$$\begin{cases} u_1(x_1, x_2, x_3, t = 0) = \cos(ax_1)\sin(ax_2)\sin(ax_3), \\ u_2(x_1, x_2, x_3, t = 0) = -\sin(ax_1)\cos(ax_2)\sin(ax_3), \\ u_3(x_1, x_2, x_3, t = 0) = 0. \end{cases} \tag{1}$$

and assuming periodic conditions in a cubic domain: $0 \leqslant x_1 \leqslant 2\pi$, $0 \leqslant x_2 \leqslant 2\pi$, $0 \leqslant x_3 \leqslant 2\pi$ with $a = 1$, the three-dimensional filtered Navier-Stokes equation

$$\begin{cases} \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \frac{1}{Re} \frac{\partial^2 u_i}{\partial x_i \partial x_j}, \\ \\ \frac{\partial u_i}{\partial x_i} = 0. \end{cases} \tag{2}$$

can be solved analytically at small times, using perturbation expansion. In (1) all quantities have been properly normalized by the initial maximum velocity magnitude U_0 in the x_1 or x_2 direction, and $L/2\pi$, where L is the physical

domain size, u_i -velocity at $i = 1, 2, 3$, corresponding to x_1, x_2, x_3 directions, $\text{Re} = LU_0/\nu$ is the Reynolds number of flow, U_0 - the characteristic velocity, $T = aU_0t, a = 2\pi/L$. The pressure p has been normalized by ρU_0^2 . Taylor and Green obtained a perturbation expansion of the velocity field, up to $O(t^5)$. The resulting average kinetic energy is:

$$E_k = \frac{U_0^2}{8} u^{'2} \tag{3}$$

where

$$u^{'2} = 1 - \frac{6T}{\text{Re}} + \frac{18T^2}{\text{Re}^2} - \left(\frac{5}{24} + \frac{36}{\text{Re}^2}\right) \frac{T^3}{\text{Re}} + \left(\frac{5}{2\text{Re}^2} + \frac{54}{\text{Re}^4}\right) T^4 -$$
$$- \left(\frac{5}{44.12} + \frac{367}{24\text{Re}^2} + \frac{4.81}{5\text{Re}^4}\right) \frac{T^5}{\text{Re}} + \left(\frac{361}{44.32} + \frac{761}{12\text{Re}^2} + \frac{324}{5\text{Re}^4}\right) \frac{T^6}{\text{Re}^2}. \tag{4}$$

The dissipation rate is written in the following form:

$$W = \mu \frac{3U_0^2 a^2}{4} W^{'} \tag{5}$$

where

$$W^{'} = \left(\frac{5}{48} + \frac{18T^2}{\text{Re}^2}\right) T^2 - \left(\frac{5}{3} + \frac{36}{\text{Re}^2}\right) \frac{T^3}{\text{Re}} + 1 - \frac{6T}{\text{Re}} +$$
$$+ \left(\frac{50}{99.64} + \frac{1835}{9.16\text{Re}^2} + \frac{54}{\text{Re}^4}\right) T^4 - \left(\frac{361}{44.32} + \frac{761}{12\text{Re}^2} + \frac{324}{5\text{Re}^4}\right) \frac{T^5}{\text{Re}}. \tag{6}$$

Numerical method for time-dependent three-dimensional flows

In this paper, we solve the three-dimensional non-stationary Navier-Stokes equations using a splitting scheme that consists of three steps [5]. In the first step, the equation for motion is solved, without taking pressure into account. For approximation of the convective and diffusion terms of the intermediate velocity field a finite-difference method in combination with cyclic penta-diagonal matrix is used [6,7], which allowed to increase the order of accuracy in space.

The intermediate velocity field is solved by using the Adams-Bashforth scheme in combination with a five-point sweep method. Let's consider the velocity component u_1 in the horizontal direction at the spatial location $(i + 1/2, j, k)$:

$$\frac{\partial u_1}{\partial t} + \frac{\partial (u_1 u_1)}{\partial x_1} + \frac{\partial (u_1 u_2)}{\partial x_2} + \frac{\partial (u_1 u_3)}{\partial x_3} = \frac{1}{\text{Re}} \left(\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_1}{\partial x_3^2}\right) \tag{7}$$

When using the explicit Adams-Bachfort scheme for convective terms and the implicit Crank-Nicholson scheme for viscous terms, equation (7) takes the

form:

$$\widehat{u1}_{i+\frac{1}{2},j,k}^{n+1} - u1_{i+\frac{1}{2},j,k}^{n} = -\frac{3dt}{2}\left[hx\right]_{i+\frac{1}{2},j,k}^{n} + \frac{dt}{2}\left[hxp\right]_{i+\frac{1}{2},j,k}^{n-1} + dt\left[ax\right]_{i+\frac{1}{2},j,k}^{n} +$$
$$+ \frac{dt}{2}\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \widehat{u1}}{\partial x_1^2}\right)_{i+\frac{1}{2},j,k}^{n+1} + \frac{dt}{2}\frac{1}{\mathrm{Re}}\left[\left(\frac{\partial^2 \widehat{u1}}{\partial x_2^2}\right)_{i+\frac{1}{2},j,k}^{n+1} + \left(\frac{\partial^2 \widehat{u1}}{\partial x_3^2}\right)_{i+\frac{1}{2},j,k}^{n+1}\right] \qquad (8)$$

where

$$\left[hx\right]_{i+\frac{1}{2},j,k}^{n} = \left(\frac{\partial u_1 u_1}{\partial x_1}\right)_{i+\frac{1}{2},j,k}^{n} + \left(\frac{\partial u_1 u_2}{\partial x_2}\right)_{i+\frac{1}{2},j,k}^{n} + \left(\frac{\partial u_1 u_3}{\partial x_3}\right)_{i+\frac{1}{2},j,k}^{n}$$

$$\left[hxp\right]_{i+\frac{1}{2},j,k}^{n-1} = \left(\frac{\partial u_1 u_1}{\partial x_1}\right)_{i+\frac{1}{2},j,k}^{n-1} + \left(\frac{\partial u_1 u_2}{\partial x_2}\right)_{i+\frac{1}{2},j,k}^{n-1} + \left(\frac{\partial u_1 u_3}{\partial x_3}\right)_{i+\frac{1}{2},j,k}^{n-1}$$

$$\left[ax\right]_{i+\frac{1}{2}jk}^{n} = \frac{1}{2}\cdot\frac{1}{\mathrm{Re}}\left[\left(\frac{\partial^2 u_1}{\partial x_1^2}\right)_{i+\frac{1}{2}jk}^{n} + \left(\frac{\partial^2 u_1}{\partial x_2^2}\right)_{i+\frac{1}{2}jk}^{n} + \left(\frac{\partial^2 u_1}{\partial x_3^2}\right)_{i+\frac{1}{2}jk}^{n}\right]$$

Discretization of convective terms looks as [8]:

$$\left(\frac{\partial u_1 u_1}{\partial x_1}\right)\bigg|_{i+\frac{1}{2},j,k} = -\frac{1}{24\Delta x_1}(u_1^2)_{i+2,j,k} + \frac{9}{8\Delta x_1}\left[(u_1^2)_{i+1,j,k} - (u_1^2)_{i,j,k}\right] +$$
$$+ \frac{1}{24\Delta x_1}(u_1^2)_{i-1,j,k} + O(\Delta x_1^4);$$

$$\left(\frac{\partial u_1 u_2}{\partial x_1}\right)\bigg|_{i+\frac{1}{2},j,k} = \frac{1}{24\Delta x_2}(u_1 u_2)_{i+\frac{1}{2},j,k} - \frac{1}{24\Delta x_2}(u_1 u_2)_{i+\frac{1}{2},j+\frac{3}{2},k} +$$
$$+ \frac{9}{8\Delta x_2}\left[-(u_1 u_2)_{i+\frac{1}{2},j-\frac{1}{2},k} + (u_1 u_2)_{i+\frac{1}{2},j+\frac{1}{2},k}\right] + O(\Delta x_2^4);$$

$$\left(\frac{\partial u_1 u_3}{\partial x_3}\right)\bigg|_{i+\frac{1}{2},j,k} = \frac{1}{24\Delta x_3}(u_1 u_3)_{i+\frac{1}{2},j,k-\frac{3}{2}} - \frac{1}{24\Delta x_3}(u_1 u_3)_{i+\frac{1}{2},j,k+\frac{3}{2}} +$$
$$+ \frac{9}{8\Delta x_3}\left[-(u_1 u_3)_{i+\frac{1}{2},j,k-\frac{1}{2}} + (u_1 u_3)_{i+\frac{1}{2},j,k+\frac{1}{2}}\right] + O(\Delta x_3^4);$$

where

$$(u_1 u_1)_{i,j,k} = \frac{1}{256}\left(9u_{1i-\frac{1}{2},j,k} + 9u_{1i+\frac{1}{2},j,k} - u_{1i+\frac{3}{2},j,k} - u_{1i-\frac{3}{2},j,k}\right)^2;$$

$$(u_1 u_2)_{i+\frac{1}{2},j+\frac{1}{2},k} = \frac{1}{256}\left(9u_{1i+\frac{1}{2},j,k} + 9u_{1i+\frac{1}{2},j+1,k} - u_{1i+\frac{1}{2},j+2,k} - u_{1i+\frac{1}{2},j-1,k}\right)$$
$$\cdot\left(9u_{2i+1,j+\frac{1}{2},k} + 9u_{2i,j+\frac{1}{2},k} - u_{2i+2,j+\frac{1}{2},k} - u_{2i-1,j+\frac{1}{2},k}\right);$$

$$(u_1 u_3)_{i+\frac{1}{2},j,k+\frac{1}{2}} = \frac{1}{256}\left(-u_{1i+\frac{1}{2},j,k+2} - u_{1i+\frac{1}{2},j,k-1} + 9u_{1i+\frac{1}{2},j,k} + 9u_{1i+\frac{1}{2},j,k+1}\right)$$
$$\cdot\left(-u_{3i+2,j,k+\frac{1}{2}} + 9u_{3i+1,j,k+\frac{1}{2}} + 9u_{3i,j,k+\frac{1}{2}} - u_{3i-1,j,k+\frac{1}{2}}\right);$$

Then the left hand side of equation (8) is denoted by $q_{i+\frac{1}{2}jk}$

$$q_{i+\frac{1}{2}jk} \equiv \widehat{u_1}_{i+\frac{1}{2},j,k}^{n+1} - u_{1}{}_{i+\frac{1}{2},j,k}^{n}. \tag{9}$$

We find $\widehat{u_1}_{i+\frac{1}{2}jk}^{n+1}$ from equation (9)

$$\widehat{u_1}_{i+\frac{1}{2},j,k}^{n+1} = q_{i+\frac{1}{2}jk} + u_{1}{}_{i+\frac{1}{2},j,k}^{n}.$$

Replacing all $\widehat{u_1}_{i+\frac{1}{2},j,k}^{n+1}$ from the equations (8) we obtain

$$-\frac{dt}{2}\frac{1}{\text{Re}}\left(\frac{\partial^2 q}{\partial x_1^2}\right)_{i+\frac{1}{2},j,k} - \frac{dt}{2}\frac{1}{\text{Re}}\left(\frac{\partial^2 q}{\partial x_2^2}\right)_{i+\frac{1}{2},j,k} - \frac{dt}{2}\frac{1}{\text{Re}}\left(\frac{\partial^2 q}{\partial x_3^2}\right)_{i+\frac{1}{2},j,k} +$$
$$+ q_{i+\frac{1}{2}jk} = -\frac{3dt}{2}\left[hx\right]_{i+\frac{1}{2},j,k}^{n} + \frac{dt}{2}\left[hxp\right]_{i+\frac{1}{2},j,k}^{n-1} + 2dt\left[ax\right]_{i+\frac{1}{2},j,k}^{n} \tag{10}$$

We can re-write equation (10) as

$$\left[1 - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_1^2} - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_2^2} - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_3^2}\right] q_{i+\frac{1}{2}jk} = d_{i+\frac{1}{2},j,k}, \tag{11}$$

where

$$d_{i+\frac{1}{2}jk} = -\frac{3dt}{2}\left[hx\right]_{i+\frac{1}{2},j,k}^{n} + \frac{dt}{2}\left[hxp\right]_{i+\frac{1}{2},j,k}^{n-1} + 2\cdot dt\left[ax\right]_{i+\frac{1}{2},j,k}^{n}.$$

Assuming that equation (11) has the second-order accuracy in time, we may solve the following equation instead:

$$\left[1 - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_1^2}\right]\left[1 - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_2^2}\right]\left[1 - \frac{dt}{2}\frac{1}{\text{Re}}\frac{\partial^2}{\partial x_3^2}\right] q_{i+\frac{1}{2},j,k}^{*} = d_{i+\frac{1}{2},j,k} \tag{12}$$

We can show that Equation (12) is an $O(\Delta t^4)$ approximation to equation (11) [5].

Equation (12) is a factorization approximation to equation (11), which allows each spatial direction to be treated sequentially. If we denote the solution to Equation (12) as $q_{i+\frac{1}{2}jk}^{*}$, by expanding Equation (12), subtracting equation (11) from it, and noting that $q_{i+\frac{1}{2}jk} \sim \mathcal{O}\left(dt^2\right)$, we obtain $\left(q_{i+\frac{1}{2}jk}^{*} - q_{i+\frac{1}{2}jk}\right) \sim \mathcal{O}\left(dt^4\right)$. Therefore, Equation (12) is actually an order $\mathcal{O}\left(dt^4\right)$ approximation to equation (11), rather than an order $\mathcal{O}\left(dt^3\right)$ approximation as stated in [5] without proof. Since the difference between $q_{i+\frac{1}{2}jk}^{*}$ and $q_{i+\frac{1}{2}jk}$ is of higher order, we shall return to the same notation and just use $q_{i+\frac{1}{2}jk}$.

To determine $q_{i+\frac{1}{2}jk}$, equation (12) is solved in 3 stages in sequence as follows:

$$\left[1 - \frac{dt}{2}\cdot\frac{1}{\text{Re}}\cdot\frac{\partial^2}{\partial x_1^2}\right] A_{i+\frac{1}{2},j,k} = d_{i+\frac{1}{2},j,k}; \tag{13}$$

$$\left[1 - \frac{dt}{2} \cdot \frac{1}{\mathrm{Re}} \cdot \frac{\partial^2}{\partial x_2^2}\right] B_{i+\frac{1}{2},j,k} = A_{i+\frac{1}{2},j,k}; \qquad (14)$$

$$\left[1 - \frac{dt}{2} \cdot \frac{1}{\mathrm{Re}} \cdot \frac{\partial^2}{\partial x_3^2}\right] q_{i+\frac{1}{2},j,k} = B_{i+\frac{1}{2}jk}. \qquad (15)$$

At the first stage, $A_{i+\frac{1}{2},j,k}$ is sought in the coordinate direction x_1:

$$\left[1 - \frac{dt}{2} \cdot \frac{1}{\mathrm{Re}} \cdot \frac{\partial^2}{\partial x_1^2}\right] A_{i+\frac{1}{2},j,k} = d_{i+\frac{1}{2},j,k},$$

$$A_{i+\frac{1}{2},j,k} - \frac{dt}{2} \cdot \frac{1}{\mathrm{Re}} \cdot \left(\frac{\partial^2 A}{\partial x_1^2}\right)_{i+\frac{1}{2},j,k} = d_{i+\frac{1}{2},j,k},$$

$$A_{i+\frac{1}{2},j,k} - \frac{dt}{2}\frac{1}{\mathrm{Re}}\frac{-A_{i+\frac{5}{2},j,k} + 16A_{i+\frac{3}{2},j,k} - 30A_{i+\frac{1}{2},j,k}}{12\Delta x_1^2} +$$
$$+ \frac{16A_{i-\frac{1}{2},j,k} - A_{i-\frac{3}{2},j,k}}{12\Delta x_1^2} = d_{i+\frac{1}{2},j,k}, \quad (16)$$

$$s_1 A_{i+\frac{5}{2},j,k} - 16 s_1 A_{i+\frac{3}{2},j,k} + (1 + 30 s_1) A_{i+\frac{1}{2},j,k} -$$
$$- 16 s_1 A_{i-\frac{1}{2},j,k} + s_1 A_{i-\frac{3}{2},j,k} = d_{i+\frac{1}{2},j,k}, \quad (17)$$

where $s_1 = \frac{dt}{24 \cdot \mathrm{Re} \cdot \Delta x_1^2}$.

This equation (17) is solved by the cyclic penta-diagonal matrix method, which yields $A_{i+\frac{1}{2}jk}$.

The same procedure is repeated next for the x_2 directions in the second stage, namely, $B_{i+\frac{1}{2}jk}$ is obtained by solving equation (14), with the solution from the first stage as the coefficient on the right hand and the coefficient s_1 in the penta-diagonal matrix replaced by $s_2 = \frac{dt}{24 \cdot \mathrm{Re} \cdot \Delta x_2^2}$. Finally, in the third stage, $q_{i+\frac{1}{2}jk}$ is solved through the similar penta-diagonal system shown in equation (15). Once we have determined the value of $q_{i+\frac{1}{2},j,k}$, we find $\widehat{u1}_{i+\frac{1}{2},j,k}^{n+1}$

$$\widehat{u1}_{i+\frac{1}{2},j,k}^{n+1} = q_{i+\frac{1}{2},j,k} + u1_{i+\frac{1}{2},j,k}^{n}.$$

The velocity components $\widehat{u2}_{i,j+\frac{1}{2},k}^{n+1}$ and $\widehat{u3}_{i,j,k+\frac{1}{2}}^{n+1}$ are solved in a similar manner.

In the second step, the pressure Poisson equation is solved, which ensures that the continuity equation is satisfied. The Poisson equation is transformed from the physical space into the spectral space by using a Fourier transform. The resulting intermediate velocity field does not satisfy the continuity equation. The final velocity field is obtained by adding to the intermediate field the term corresponding to the pressure gradient:

$$u_1^{n+1} = \widehat{u_1}^{n+1} - dt \frac{\partial p}{\partial x_1};$$

$$u_2^{n+1} = \widehat{u_2}^{n+1} - dt \frac{\partial p}{\partial x_2};$$

$$u_3^{n+1} = \widehat{u_3}^{n+1} - dt \frac{\partial p}{\partial x_3}.$$

Substituting the continuity equation, we obtain the Poisson equation for the pressure field:

$$\frac{\partial^2 p}{\partial x_1^2} + \frac{\partial^2 p}{\partial x_2^2} + \frac{\partial^2 p}{\partial x_3^2} = dt \left(\frac{\partial \widehat{u_1}^{n+1}}{\partial x_1} + \frac{\partial \widehat{u_2}^{n+1}}{\partial x_2} + \frac{\partial \widehat{u_3}^{n+1}}{\partial x_3} \right) \equiv F_{i,j,k},$$

where $F_{i,j,k}$ denotes the known right hand side of the Poisson equation, with each term approximated by an order $O(\Delta x^4)$ finite-difference approximation. For example, the first term in $F_{i,j,k}$ is approximated as

$$dt \frac{\partial \widehat{u_1}^{n+1}}{\partial x_1} \bigg|_{i+\frac{1}{2},j,k} = dt \frac{\widehat{u_1}_{i-\frac{3}{2},j,k}^{n+1} - 8\widehat{u_1}_{i-\frac{1}{2},j,k}^{n+1}}{12\Delta x_1} + \frac{8\widehat{u_1}_{i+\frac{3}{2},j,k}^{n+1} - \widehat{u_1}_{i+\frac{5}{2},j,k}^{n+1}}{12\Delta x_1} + O(\Delta x_1^4).$$

To be consistent with the spatial accuracy in the first step, the left hand side of the above Poisson equation is discretized using 5-point scheme of $O(\Delta x^4)$ accuracy, as follows:

$$\left[\frac{-P_{i+2,j,k} + 16P_{i+1,j,k} - 30P_{i,j,k} + 16P_{i-1,j,k} - P_{i-2,j,k}}{12\Delta x_1^2} \right] +$$

$$+ \left[\frac{-P_{i,j+2,k} + 16P_{i,j+1,k} - 30P_{i,j,k} + 16P_{i,j-1,k} - P_{i,j-2,k}}{12\Delta x_2^2} \right] + \quad (18)$$

$$+ \left[\frac{-P_{i,j,k+2} + 16P_{i,j,k+1} - 30P_{i,j,k} + 16P_{i,j,k-1} - P_{i,j,k-2}}{12\Delta x_3^2} \right] = F_{i,j,k}.$$

Now we apply the three dimensional Fourier transform

$$P_{i,j,k} = \frac{1}{N} \sum_{m=0}^{N_1-1} \sum_{n=0}^{N_2-1} \sum_{s=0}^{N_3-1} V_1^{im} V_2^{jn} V_3^{sk} \cdot \widehat{p}_{m,n,s};$$

$$F_{i,j,k} = \frac{1}{N} \sum_{m=0}^{N_1-1} \sum_{n=0}^{N_2-1} \sum_{s=0}^{N_3-1} V_1^{im} V_2^{jn} V_3^{sk} \cdot \widehat{f}_{m,n,s}. \quad (19)$$

The inverse transforms are:

$$\widehat{p}_{m,n,s} = \frac{1}{N} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} \sum_{k=0}^{N_3-1} V_1^{-im} V_2^{-jn} V_3^{-sk} \cdot P_{i,j,k};$$

$$\widehat{f}_{m,n,s} = \frac{1}{N} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} \sum_{k=0}^{N_3-1} V_1^{-im} V_2^{-jn} V_3^{-sk} \cdot F_{i,j,k}.$$

(20)

where $N = N_1 \cdot N_2 \cdot N_3$, $V_1 = e^{\iota \left(\frac{2\pi}{N_1} \right)}$, $V_2 = e^{\iota \left(\frac{2\pi}{N_2} \right)}$, and $V_3 = e^{\iota \left(\frac{2\pi}{N_3} \right)}$.

Substituting equation (19) into equation (18), we obtain quickly the solution for the pressure field in the spectral space as

$$\widehat{p}_{m,n,s} = \frac{12\widehat{f}_{m,n,s}}{Q_1 + Q_2 + Q_3} \tag{21}$$

where

$$Q_1 = \frac{1}{\Delta x_1^2} \left[-2\cos\left(\frac{4\pi m}{N_1}\right) + 32\cos\left(\frac{2\pi m}{N_1}\right) - 30 \right],$$

$$Q_2 = \frac{1}{\Delta x_2^2} \left[-2\cos\left(\frac{4\pi n}{N_2}\right) + 32\cos\left(\frac{2\pi n}{N_2}\right) - 30 \right],$$

$$Q_3 = \frac{1}{\Delta x_3^2} \left[-2\cos\left(\frac{4\pi s}{N_3}\right) + 32\cos\left(\frac{2\pi s}{N_3}\right) - 30 \right].$$

An inverse Fourier transform is then performed to obtain the pressure $P_{i,j,k}$ in the physical space. The obtained pressure field is then used at the third step to determine the final velocity field.

Calculation of turbulence characteristics for the Taylor-Green vortex problem

To determine the turbulent flow characteristics in the physical space, it is necessary to average different values in the whole domain. The averaged values will be used to find the turbulent characteristics. The value averaged over the entire domain is calculated by the following formula [9]:

$$< u_i > = \frac{1}{N_1 N_2 N_3} \sum_{n=1}^{N_1} \sum_{m=1}^{N_2} \sum_{q=3}^{N_3} (\bar{u_i})_{n,m,q}.$$

The dissipation rate is calculated by the following formula:

$$\epsilon = < 2v S_{ij} S_{ij} > = 2v \left[\left\langle \left(\frac{\partial u_1}{\partial x_1} \right)^2 \right\rangle + \left\langle \left(\frac{\partial u_2}{\partial x_2} \right)^2 \right\rangle + \left\langle \left(\frac{\partial u_3}{\partial x_3} \right)^2 \right\rangle \right] +$$

$$+ 2v \left[\frac{1}{2} \left\langle \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right)^2 \right\rangle + \frac{1}{2} \left\langle \left(\frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1} \right)^2 \right\rangle + \frac{1}{2} \left\langle \left(\frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \right)^2 \right\rangle \right]$$

The turbulent kinematic energy is found in the following way:

$$E_k = \frac{1}{2}\left(\langle u_1^2\rangle + \langle u_2^2\rangle + \langle u_3^2\rangle\right) = \frac{3}{2}\left\langle u_1^2\right\rangle.$$

Velocity derivative skewness is defined in the following form:

$$S(t) = \frac{1}{3}\left[\frac{\langle(\partial u_1)^3\rangle + \langle(\partial u_2)^3\rangle + \langle(\partial u_3)^3\rangle}{\langle(\partial u_1)^2\rangle^{\frac{2}{3}} + \langle(\partial u_2)^2\rangle^{\frac{2}{3}} + \langle(\partial u_3)^2\rangle^{\frac{2}{3}}}\right].$$

Numerical results

The numerical model allowed to describe the turbulence decay based on the hybrid method, using finite-difference method in combination with cyclic penta-diagonal for solving equation of motion and spectral method for solution of the Poisson equation.

For this task, the characteristic values of the velocity and length were taken equal to $U_0 = 1$, and $L = 1$, respectively. The grid resolution was set to $128 \times 128 \times 128$. The Reynolds numbers $\mathrm{Re} = \frac{U_0 L}{v(2\pi)}$ were taken: 1) $\mathrm{Re} = 100$; 2) $\mathrm{Re} = 300$; 3) $\mathrm{Re} = 600$, $L_B = 1$. $T = aU_0 t = \frac{2\pi U_0 t}{L_B}$, where $a = \frac{2\pi}{L_B} = 2\pi$. Initial values for $k(t=0) = 0.125d0$ and $\epsilon(t=0) = \frac{0.75}{2\pi\,\mathrm{Re}}\left(\frac{2\pi}{L_B}\right)^2$.

In Tables 1 to 3, a qualitative comparative analysis for kinetic energy and dissipation rate at Reynolds numbers: 1) $\mathrm{Re} = 100$; 2) $\mathrm{Re} = 300$; 3) $\mathrm{Re} = 600$ were made. It is found, that the average errors of kinetic energy between analytical Taylor Green vortex problem solutions and numerical simulation result is $\varepsilon_{err}(kinetic\,energy) = 10^{-4}$ at different Reynolds numbers. The dissipation rate has average errors $\varepsilon_{err}(dissipation\,rate) = 10^{-2}$ between analytical and numerical solutions.

The simulation at different Reynolds numbers were compared with the analytical solution of Taylor - Green vortex problem in terms of: average kinetic energy and average dissipation rate of turbulent flow. Fig. 1 presents the comparison of the average turbulent kinetic energy obtained in this work with the analytical solution of Taylor-Green vortex problem at different Reynolds numbers: 1) $\mathrm{Re} = 100$; 2) $\mathrm{Re} = 300$; 3) $\mathrm{Re} = 600$. The obtained results show the satisfactory agreement up to $T = 3$ for average turbulent kinetic energy. The error between analytical and numerical solutions for average kinetic energy was defined as $Error(E_k) = |E_k^{FDM} - E_k^{TG}| = 10^{-4}$.

Fig. 2 compares the matching results of the dissipation rate of turbulence decay over the time received in the numerical simulation and from the exact solution of the Taylor-Green vortex problem at different Reynolds numbers. We observe a good agreement till $T = 1.5$ because the order of accuracy in time is $O(t^3)$. In the simulation results, the average error between analytical and numerical solutions for dissipation rate was equal to $\varepsilon_{err}(diss.\,rate) = |\varepsilon_{err}(FDM) - \varepsilon_{err}(TG)| = 10^{-2}$.
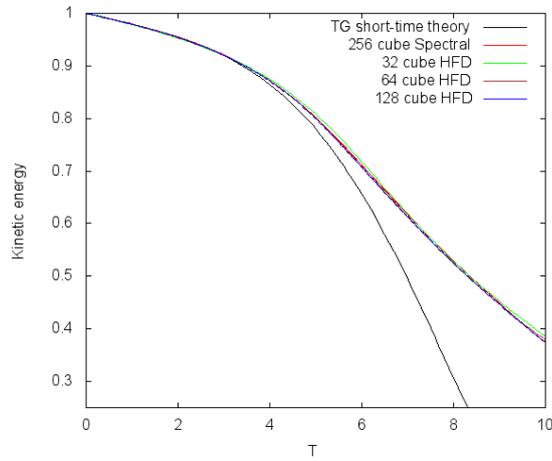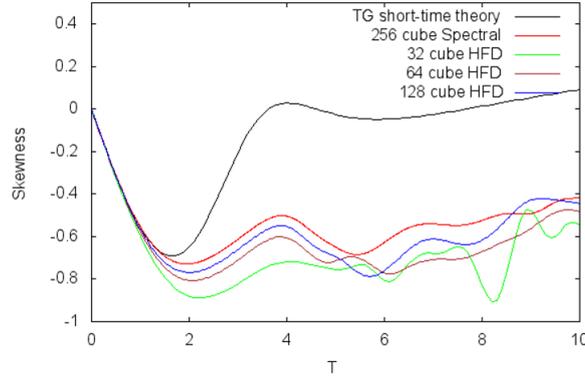
Figure 1. Comparison simulation result of the evolution of kinetic energy over the time with direct methods of Taylor - Green vortex at different Reynolds number
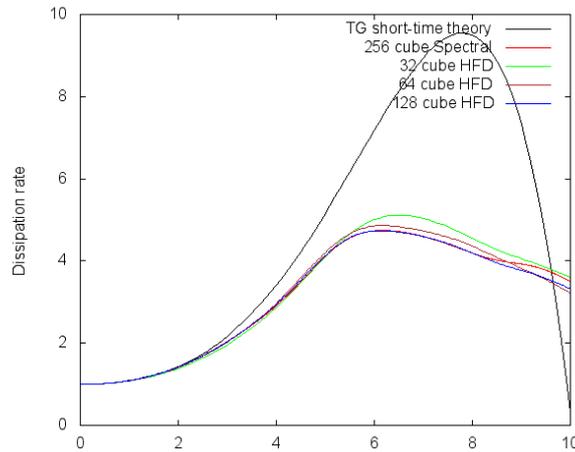
Figure 2. Comparison simulation result of the dissipation rate of turbulence decay over the time with direct methods of Taylor - Green vortex at different Reynolds number.

Figure 3 compares the results of the comparison of the evolution of skewness with respect to time, obtained from numerical simulation, and the analytical solution of the Taylor-Green vortex problem at $Re = 300$. It can be seen from Fig. 3 that with increasing resolution of the computational grid the results of turbulence skewness of the hybrid method tends to exponential results of the pseudospectral method for the computational grid 256^3.

Figure 3. Comparison simulation result of the evolution of skewness over the time with direct methods of Taylor - Green vortex at Re=300

Table 1. Qualitative comparison for kinetic energy and dissipation rate at Re=100

Time	Kinetic energy		
T	TG, Re=100	FDM, Re=100	error, Re=100
0	0,125	0,124999985	1,50E-08
0,25	0,123135023	0,123119563	1,55E-05
0,5	0,121274717	0,121250503	2,42E-05
0,75	0,119397029	0,119370982	2,60E-05
1	0,117480934	0,117458388	2,25E-05
1,25	0,115506172	0,115490906	1,53E-05
1,5	0,113453701	0,113448143	5,56E-06
1,75	0,111306652	0,111311972	5,32E-06
2	0,109051958	0,10906744	1,55E-05

Time	Dissipation rate		
T	TG, Re=100	FDM, Re=100	error, Re=100
0	1	1	0,00E+00
0,25	0,991395	0,991305	9,04E-05
0,5	0,994895	0,99482	7,52E-05
0,75	1,009837	1,009854	1,76E-05
1	1,035803	1,035941	1,38E-04
1,25	1,072311	1,072576	2,65E-04
1,5	1,118506	1,118979	4,72E-04
1,75	1,172854	1,173887	1,03E-03
2	1,232831	1,235449	2,62E-03

Conclusion

A numerical algorithm for solving non-stationary three-dimensional Navier-Stokes equations has been developed to study the time evolution of a turbulent

Table 2. Qualitative comparison for kinetic energy and dissipation rate at Re=300

Time	Kinetic energy		
T	TG, Re=300	FDM, Re=300	error, Re=300
0	0,125	0,124999985	1,50E-08
0,25	0,124375217	0,124355979	1,92E-05
0,5	0,123745479	0,123708762	3,67E-05
0,75	0,123102576	0,123051584	5,10E-05
1	0,122437842	0,122375496	6,23E-05
1,25	0,121741787	0,121670842	7,09E-05
1,5	0,121003874	0,120927364	7,65E-05
1,75	0,120212354	0,120134518	7,78E-05
2	0,119354106	0,119281456	7,27E-05

Time	Dissipation rate		
T	TG, Re=300	FDM, Re=300	error, Re=300
0	1	1	0,00E+00
0,25	1,001467	1,001291	1,76E-04
0,5	1,015872	1,015477	3,95E-04
0,75	1,0437	1,042984	7,16E-04
1	1,085986	1,08465	1,34E-03
1,25	1,144216	1,141567	2,65E-03
1,5	1,22023	1,214906	5,32E-03
1,75	1,316116	1,305717	1,04E-02
2	1,434116	1,414707	1,94E-02

Table 3. Qualitative comparison for kinetic energy and dissipation rate at Re=600

Time	Kinetic energy		
T	TG, Re=600	FDM, Re=600	error, Re=600
0	0,125	0,124999985	1,50E-08
0,25	0,12468721	0,124667168	2,00E-05
0,5	0,124371126	0,124331363	4,00E-05
0,75	0,124047518	0,123990014	5,80E-05
1	0,12371181	0,123638444	7,30E-05
1,25	0,123358868	0,123271443	8,70E-05
1,5	0,122982867	0,122883387	9,90E-05
1,75	0,122577086	0,122468509	1,10E-04
2	0,122133784	0,122020796	1,10E-04

flow. The algorithm is a hybrid method combining the finite-difference and spectral methods. It is also computationally efficient. The finite-difference method in combination with cyclic penta-diagonal matrix for solving the Navier-Stokes equations allowed to reach a 4th-order accuracy in space and 3rd-order

Time	Dissipation rate		
T	TG, Re=600	FDM, Re=600	error, Re=600
0	1	1	0,00E+00
0,25	1,004001	1,003804	1,97E-04
0,5	1,021189	1,020711	4,78E-04
0,75	1,052356	1,051449	9,07E-04
1	1,098938	1,097241	1,70E-03
1,25	1,16296	1,15969	3,27E-03
1,5	1,246994	1,24067	6,32E-03
1,75	1,354102	1,342202	1,19E-02
2	1,487788	1,466306	2,15E-02

accuracy in time. The spectral method to solve the Poisson equation has high computational efficiency due to the use of a Fast Fourier transform library.

For validation of the deveoped algorithm the classical problem of Taylor and Green with the same initial flow was considered, to study the decay of the flow kinetic energy and time evolution of the viscous dissipation. The average normalized errors between analytical and numerical solutions for kinetic energy and dissipation rate were founded to be $\varepsilon_{err}(kinetic\,energy) = 10^{-4}$ and $\varepsilon_{err}(dissipation\,rate) = 10^{-2}$, respectively. Thus, the numerical simulation results of turbulence characteristics show very good agreements with analytical solution.

References

1. Taylor, G.I. and Green, A.E.: Mechanism of production of small eddies from large ones.Proceedings of the royal society, Mathematics and physical sciences, vol.158(895), pp.499–521. (1937)
2. Brachet, M.E. and Meiron, D. I. and Orszag, S. A. and Nickel, B. G., and Morf , R.H. and Frisch, U.: Small-scale structure of the Taylor-Green vortex. J.Fluid Mechanics, vo.130, pp.411-452.(1983)
3. Brachet, M.E.: Direct simulation of three - dimensional turbulence in the Taylor - Green vortex. J.Fluid Dynamics Research, vol.8, pp.1-8. (1991)
4. DeBonis, J.R.: Solutions of the Taylor-Green vortex problem using high - resolution explicit finite difference methods. J.National Aeronautics and Space Administration Technical Memorandum, vol. 2013-217850, pp. 1-28. (2013)
5. Kim, J. and Moin, P.: Application of a fractional - step method to incompressible Navier- Stokes equations. J.Computational Physics, vol.59, pp.308-323.(1985)
6. Batista, M.: A Method for Solving Cyclic Block Penta-diagonal Systems of Linear Equations. Mathematical Software, http://arxiv.org/abs/0803.0874. (2008)
7. Navon, M.: Pent: A periodic pentadiagonal systems solve. J.Communications in applied numerical methods,vol.3, pp. 63-69. (1987)
8. Kampanis, N.A. and J.A.Ekaterinaris: A staggered grid, high-order accurate method for the incompressible Navier-Stokes equations. J.Computational Physics, vol.215, pp. 589-613.(2006)

9. Zhakebayev, D.B. and Zhumagulov, B.T. and A.U. Abdibekova: The decay of MHD turbulence depending on the conductive properties of the environment. J.Magnetohydrosynamics,vol.50, No.2, pp. 121-138.(2014)

Development of technological vision system for tracing a physical object with the use of the meanshift algorithm

A. Baidildina, O. Baklanova, S. Grigoryeva, O. Shvets, and G. Shakharova

D. Serikbayev East Kazakhstan State Technical University,
Ust-Kamenogorsk, Kazakhstan
azhaparova@ektu.kz

Abstract Due to the increasing complexity of the scientific and technical problems, automatic processing and analysis of visual information are becoming more and more important issues. These technologies are used in highly demanded areas of science and technology, such as process automation, increased productivity, improved product quality, control of production equipment, intelligent robotic systems, control systems for moving vehicles, biomedical research and many others.

In this work, object tracking is implemented in the software package of the industrial robot control system using the MeanShift segmentation algorithm.

Currently, various tracking algorithms have been developed, which are modifications of the MeanShift, Kalman, and particle filters. One of the main problems in applying these algorithms in practice is the loss of the tracker object: after the object leaves the frame or comes behind some obstacle, the tracker continues to keep track, but not of the original object, but of some area on the frame that is not an object. We need the criteria for the correct operation of the tracker to control these effects. In the article, this problem was solved by imposing artificial restrictions in the form of a mask of the tracking area and the boundaries of the change in the characteristics of the object. It is given a demonstration of the system of technical vision.

Keywords: technical vision, tracking, MeanShift algorithm, image segmentation, software

Introduction

Technical vision has long been used in production automation systems to improve the quality of products and increase productivity, replacing the visual control traditionally performed by a person. The visual data obtained by solving problems with technical vision (capture and movement, object tracking, metrology, detection of defects, etc.) make it possible to improve the performance of the entire system by providing simple information "the test passed / the test is not passed"or closing the control loops.

Today, great technological progress in the field of technical vision is associated with computing power. Due to the constantly increasing processor performance and the development of parallel processing technologies, developers of vision systems can use sophisticated algorithms to visualize data and create intelligent systems.

Tracking algorithms that evaluate the positions of one or more targets on the frames of a video sequence are included in numerous applications of technical vision in robotics. The choice of image processing (tracking) method, obtained by technical means, is determined based on the nature of the image, the type of objects and assigned tasks [1, 2].

The system of visual observation consists of two main parts - the representation and localization of the target object; filtering and merging data consolidation.

The representation and localization of the target is an upward process. Usually, the computational complexity of these algorithms is rather small. This is an example of standard algorithms for implementing the representation and localization of the tracking object.

Blob tracking - the segmentation of the interior of the object.

Kernel-based tracking (Mean-shift tracking) is an iterative localization procedure based on the maximization of the similarity criterion (Bhattacharyya coefficient).

Contour tracking - search for the border of the object.

Feature matching - visual matching of features, registration.

Point feature tracking - tracking point features of the scene.

Filtering and data consolidation represents a top-down process that involves combining a priori information about a scene or an object that relates to the dynamics of an object and the computation of different hypotheses. The computational complexity of these algorithms is usually much higher. Here are examples of standard filtering algorithms:

The Kalman filter is the optimal recursive algorithm for linear functions subject to Gaussian noise.

Particle filter - useful for sampling the basic space of distribution states of nonlinear and non-Gaussian processes.

The methods of image segmentation are most commonly used for tracking objects [3]. Segmentation of images is one of the basic tasks of technical vision, since image analysis often begins with its decomposition into areas with relevant information for the task. Segmentation is usually not used on its own, but as a part of a system of technical vision, so the quality of the algorithm is evaluated based on the system as a whole. The same segmentation algorithm can be good for one task and bad for another. Thus, the choice of the segmentation method depends from the necessary problem.

Image segmentation algorithms can be divided into two classes: interactive - using custom prompts and automatic ones - not requiring user participation [4, 5]. One of the automatic image segmentation algorithms is the MeanShift algorithm. The search for the most suitable image area in the mean shift

algorithm MeanShift [6-10] begins with the area fixed in the previous frame. MeanShift procedure is introduced and described in detail in [6,7].Objectsare described using normalized color histograms, that is, the characteristic color vector in this case is a normalized color histogram. A real-valued function of similarity is constructed for histograms, based on the Bhattacharya coefficient. On the first frame of the sequence (video sequence), an object is marked in some way, and then for the next frame of the video sequence the nearest local maximum of the similarity function is determined. The MeanShift procedure proved to be effective and was further developed. However, long-term tracking cannot be produced when the proportions and sizes of the object are changed while moving, because the "non-small"window captures too much background and "slides"from the object. In [9,10] this problem was solved in special cases and a way of tracking the object, in which the dimensions change, but the proportions do not change. In [11], it is proposed to introduce a quantitative criterion for estimating the correct operation of the tracking algorithm MeanShift, which makes it possible to increase the stability of the system.

In this work, tracking the object in the industrial robot control system will be implemented in the software part based on the MeanShift segmentation algorithm. The choice of this particular segmentation algorithm among others [3, 12] is related to the following facts:

1. MeanShift segmentation algorithm has a mathematical justification (it is not heuristic).

2. MeanShift algorithm, referring to the class of clustering algorithms, does not require specifying the number of clusters, unlike other clustering algorithms used for image segmentation. This means that the MeanShift algorithm determines the number of segments of the image in the process of its operation.

3. Iterative processing of image pixels in MeanShift allows parallelization on graphic coprocessors, since the result of processing on each iteration one pixel does not depend on the result of processing the remaining pixels. Most image segmentation algorithms are computationally time-consuming, but not everyone fits into the model of massively parallel computations.

4. The good "quality"of the algorithm for segmentation of images MeanShift is evidenced by numerous publications in which MeanShift is used to solve specific problems of machine vision [13, 14, 15].

We will use the method of filtering the pixels of the image based on the color characteristics to work with the tracking module [16]. This method is characterized by the fact that histograms of color distribution are resistant to changes in scale, occlusion and rotation of the object. A moving object is tracked by filtering the pixels of the image and comparing its histogram with the position sampling histograms using the Bhattacharya distance.

Formulation of the problem

It is necessary to make the statement of the problem before the development of the algorithm. We describe the tracking process, i.e. the work of the tracker.

Let there be a video stream v representing a sequence of frames. The block number in the video sequence plays the role of the discrete time t. With the help of an expert on the first frame, an object is selected that is of interest. Quantitatively, the object is described by some set of features $(x_i, \ i = 1, ..., n)$, which is formally written as the vector $z = (x_1, \ldots x_n) \in \Omega \in \mathbb{R}^n$.

Here, $\mathbb{R}^n$ plays the role of feature space of objects in video sequence. The object's attributes can be the coordinates of the object in the frame (in this case, $z = (x_1, \ x_2)$, $n = 2$), brightness and color characteristics (luminance histograms, and RGB colors), etc. In the general n-dimensional case, tracking is performed in space of vectors-characteristics $\Omega \in \mathbb{R}^n$.

On the next frame, using the iterative algorithm, the position to which the object has moved is determined. In many tracking algorithms, the new position of the object on this frame is found using functions that allow you to compare the characteristics of different areas allocated on the frames and find the area of this frame with characteristics that are as close to the character as possible.

The evolution of an object in a video sequence is described by a change in the characteristics of the object $z(t) = (\ x_1(t), \ \ldots, \ x_n(t))$. The tracker can be considered as an algorithm that implements the mapping $\varphi : \Omega \to \Omega$ connecting the positions of the object in the feature space at different moments: $z(t+1) = \varphi(z(t))$. The evolutionary variable t in the video sequence has the meaning of the frame number. Let z_0 be the position of the object at the initial time $t = 0$, and the position of the object at time t, obtained with the help of the tracker φ, in terms of $z(t, z_0) = \varphi(t, z_0)$.

Let's imagine a mathematical model of a tracker as a dynamic system. In this case, the variable t varies continuously on some time interval, $t \in [0, \ T]$. Assuming that objects evolve independently of each other, we will write down the dynamic system describing the tracking of the object in the form

$$z(t) = f(t, \ z(t)), \tag{1}$$

where the functions $f(t, \ z) = (f_1 \ (t, z), \ \ldots, \ fn(t, z))$ are defined and continuously differentiable in the domain $\Omega \subset \mathbb{R}^n$, the dot denotes the derivative $\Omega \subset \mathbb{R}^n$ on t.

The existence and uniqueness of the solution $z(t, z_0)$ of the Cauchy problem for the dynamical system (1) with some given initial condition

$$z(t)|_{t=0} = \ z_0 \tag{2}$$

is determined by known conditions [17], they include the Lipschitz condition, which can be written in the form

$$|f_i(t, z) - f_i(t, \ z')| \le N \sum_{j=1}^{n} | \ x_j - \ x'_j \ |, \ \ z = (x_1, \ldots x_n), \ \ i = 1, \ \ldots, \ n \ , \tag{3}$$

where N - is a Lipschitz constant.

The expression $\rho(z, z') = \sum_{i=1}^{n} \mid x_i - x'_i \mid$ has the meaning of distance (metric) in the space $\Omega \subset \mathbb{R}^n$ of vectors - characteristics z.

Then the Lipschitz condition can be written in the form

$$\rho(f(t, z), f(t, z')) \leq nN \rho(z, z'). \tag{4}$$

Distances Bhattacharya were used as the metrics [18].

Object Tracking Algorithm Development

Currently, various tracking algorithms have been developed, which are modifications of the Mean-Shift, Kalman, and particle filters [2, 16]. One of the main problems in applying these algorithms in practice is the loss of the tracker object: after the object leaves the frame or comes behind some obstacle, the tracker continues to follow, but not beyond the original object, but behind some area on the frame that is not an object. We need the criteria for the correct operation of the tracker to control these effects.This problem was solved by imposing artificial restrictions in the form of a mask of the tracking area and the boundaries of the change in the characteristics of the object, such as geometric dimensions, shape, color, etc.

The mathematical model of the object on the slide frame, as noted above, is the characteristic vector computed for the area of the image containing the object (for example, the color histogram) [8]. In the work, the movement of an object is classified into a regular (motion of the object along a continuous trajectory) and irregular (discontinuities of trajectories due to obscuration of the object of tracking by other objects, jump of the object, etc.).

In the case of regular object movement, the tracker is considered as a dynamic system, which makes it possible to use the conditions of existence, uniqueness and stability of the solution of such a system as the criterion for the correct operation of the tracker.

In order to start monitoring the object, it is necessary to specify its initial position, thereby setting the model of the object [16]. As described above, there are many different ways of selecting an object. For this approach, let us use the expert allocation of the rectangular shape in which the object is located. In the tracking process, we will follow the contour [6].

Let $\{\xi_k\}_{k=1,\dots,r}$ be the coordinates of the pixels in the region that selects the object.

Coordinates coincide with the coordinates of the center of the object in the previous frame. As the vector of the features of the object, we select the color histogram of the tracking object, which, according to [19], we write in the form

$$q_u = C \sum_{l=1}^{r} k(\parallel \xi_l \parallel^2) \ \delta[b(\xi_l) - u], \ u = 1, \dots, m. \tag{5}$$

Here the function $b(\xi_1)$ is the cell number of the color histogram corresponding to the color of the pixel with the coordinates ξ_l; δ - Kronecker

delta symbol; $k(\parallel \xi_l \parallel^2)$ - is the kernel by which smaller weights are assigned to pixels farthest from the center [16], $\parallel \xi_l \parallel$ is the Euclidean norm of the vector ξ_l.

We define the kernel $K(\xi_l)$ as a real function defined on a nonnegative real semi-axis, such that $K(\xi_l) = k(\parallel \xi_l \parallel^2)$. The normalizing factor $C = \frac{1}{\sum_{l=1}^{r} k(\parallel \xi_l \parallel^2)}$ is obtained from the condition $\sum_u q_u = 1$.

During tracking, it is assumed that the current frame needs to determine the position of the object from a known position in the previous frame. On the current frame, the area of the possible position of the object (the candidate of the object) is selected, which is then refined.

Let $\{\xi_l\}_{l=1,\ldots,r}$ be the positions of the object's candidate pixels, then, following [16], the applicant's histogram is calculated by the formula

$$\widehat{p}_u(y) = C_h \sum_{l=1}^{r} k\left(\parallel \frac{y - \xi_l}{h} \parallel^2 \right) \delta[b(\xi_l) - u], \ u = 1, \ldots, r, \tag{6}$$

where C_h - is the normalization constant, h - is the window size used in constructing the histogram [16].

We will calculate the difference between the histograms by the formula proposed by Bhattacharya [18]:

$$p[p(y), q] = \sum_{u=1}^{r} \sqrt{p_u(y) q_u}, \tag{7}$$

where $u -$ is the number of the histogram cell.

It is proposed in [20] a formula for calculating the distance between histograms based on the Bhattacharya formula:

$$d[p(y), q] = \sqrt{1 - p[p(y), q]} = \sqrt{1 - \sum_{u=1}^{r} \sqrt{p_u(y) q_u}}. \tag{8}$$

Let's consider the function $\rho[\rho(y), q]$, depending on the coordinate y of the window center, in which the histogram $\rho(y)$ of the candidate of the object is calculated. We expand this function in a Taylor series in the neighborhood of the point $\rho(\widetilde{y}_0)$:

$$\rho\left[\widehat{p}_u(y), q\right] = \frac{1}{2} \sum_{u=1}^{r} \sqrt{\widehat{p}_u(\widetilde{y}_0) \widehat{q}_u} + \frac{1}{2} \sum_{u=1}^{r} \widehat{p}_u(y) \sqrt{\frac{\widehat{q}_u}{\widehat{p}_u(\widetilde{y}_0)}} \approx \tag{9}$$

$$\approx \frac{1}{2} \sum_{u=1}^{r} \sqrt{\widehat{p}_u(\widetilde{y}_0) \widehat{q}_u} + \frac{C_h}{2} \sum_{l=1}^{r} w_l k\left(\parallel \frac{y - \xi_1}{h} \parallel^2 \right)$$

where

$$\omega_l = \sum_{u=1}^{r} \sqrt{\frac{\widehat{g}_u}{\widehat{p}_u(\widetilde{y}_0)}} \delta\left[b\left(\xi_l\right) - u\right]. \tag{10}$$

It is necessary to maximize the second term in (10), varying y to maximize $\rho\left[\widehat{p}_u(y), q\right]$. This value is an estimate of the distribution of the probability density of the difference between the histogram of the object and the histogram of the candidate at the point y with the kernel $k\left(\left\|\frac{y-\xi_l}{h}\right\|\right)^2$ for data weighted with w_l. The modulus of this function of probability density distribution can be found using the MeanShift procedure. Thecandidate's area of the object in this procedure is recursively moved from the initial position of the candidate of the object $y_0 = \widetilde{y_0}$ to the new position by the formula

$$y_1 = \frac{\sum\limits_{i=1}^{n} \xi_i \omega_i g\left(\left\|\frac{y_0 - \xi_i}{h}\right\|\right)}{\sum\limits_{i=1}^{n} \omega_i g\left(\left\|\frac{y_0 - \xi_i}{h}\right\|\right)}, \tag{11}$$

where $g(x) = -k^{'}(x)$ under the condition that the derivative $k^{'}(x)$ of the kernel $k(x)$ exists for all $x \in [0, \infty)$, except for a set of measure zero.

The tracking algorithm that is implemented in software execution is as follows.

Let the object be characterized by the histogram $\{q_u\}_{u=1,...,r}$, its position on the previous frame $\widetilde{y_0}$.

1 Calculate the difference between the histogram of the candidate of the object $p(y_0)$ to find an object on the current frame and the histogram of the object q $\rho\left[p(y), q\right] = \sum_{u=1}^{r} \sqrt{\rho_u(y)q_u}$.

2 Calculate the weight w_l according to (10).

3 Find a new position y_1 by the formula (11) for the object candidate.

4 Calculate the histogram $\{p_u(y_1)\}_{u=1,...,r}$, of the window for the new position of the object and estimate the similarity of the histograms of the new object and object candidate according to the formula

$$\rho\left[p(y_1), q\right] = \sum_{u=1}^{r} \sqrt{\rho_u(y_1)q_u}. \tag{12}$$

5 While $\rho\left[p(y_1), q\right] < \rho\left[p(y_0), q\right]$ do iterations $y_1 = \frac{y_0 + y_1}{2}$. Thancalculate $\rho\left[p(y_1), q\right]$.

6 If $\parallel y_1 - y_0 \parallel < \varepsilon$, than stop, else $y_0 = y_1$ and return to step 2.

Let's present the implementation of the developed algorithm in the software part of the vision system for the robot manipulator.

Software

The vision system includes three basic operations: image acquisition, image processing and analysis, transfer of processing results to the process control system. The software is the main component of technical vision and performs the main task of processing digital information. The object-oriented approach

is based on the software design technology of the developed system of technical vision. C programming language was used for visualization.

An image capture module, an object tracking module, an analysis module have been developedin accordance with the developed structural diagram.

The vision system consists of a web camera connected to a computer, as well as a program for recognizing objects and controlling the manipulator. The manipulator has six degrees of freedom and has a controller that is connected to the computer via the COM interface. The webcam is mounted on the tripod vertically to see the work area from the top.

Figure 1 shows the interface of the developed software application. After starting the program, it is necessary to select (connect) the optical element of the image reading (video camera) from the drop-down list of the connected equipment and press the "start"button.

The user interface represents two windows. The tracking object is displayed in the left window. The tracking object must be placed correctly in accordance with the requirements for further tracking.

It is displayed the processed scene with the object marked on in the right window. To do this, click the "frame"button. The image is locked. Manually select the object with the mouse. With the processed image, further analysis (tracking) will take place. After pressing the "start"button, the tracking module starts working.
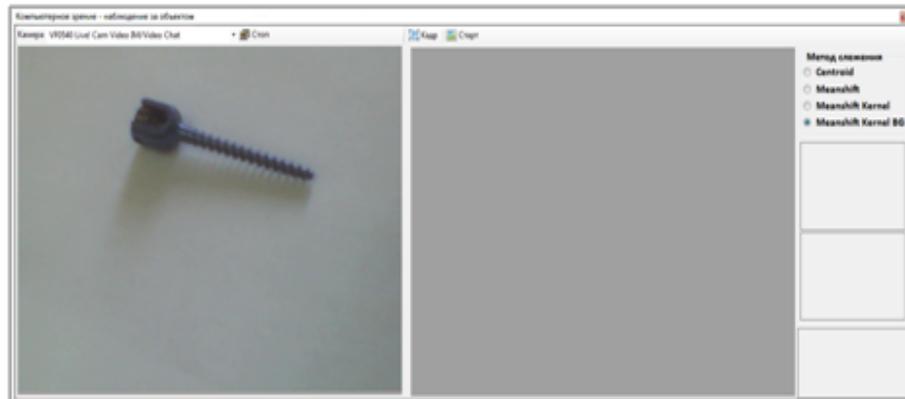

Figure 1. Object of tracing

Figure 2 shows an example of how the program module of the technical vision system works. At the first stage, we see a real-time tracking object.

If the object is positioned correctly in the initial position and satisfies the requirements for fixing it to a reference one, then proceed to the second stage. Press the "frame"button and fix the tracking object.

The next step is to segment the image object. For this, we select it with a mouse and analyze it. A histogram of the color distribution of the reference object is constructed according to the developed pixel filtering algorithm based on the color characteristics.

Information about the object is stored and used for comparison and analysis while tracking the parameters of the object in time, i.e. with its possible change.
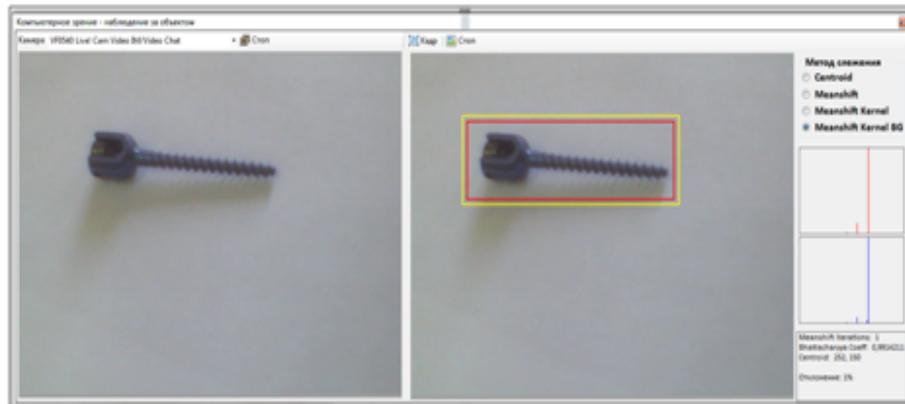

Figure 2. Fixing and analysis of the tracking object

In more detail, let's look at the description of the workspace of the software module, where graphical analysis results of the technical tracking system are displayed (Figure 3).

The user is offered to select the Meanshift segmentation algorithms implemented in the program module on the panel. The first color distribution histogram, shown in red in the example, defines a fixed tracking object, and is a normal (standard) color histogram. This information is retained for comparative analysis and calculation of the deviation factor. The position of the object can change in time, which is fixed by the camera and processed by the program module.

In accordance with the change in the object, it is generated a set of pixels of the object image based on the color characteristics. This procedure is displayed in the form of a histogram, shown in blue. The panel displays the iteration values of the average MeanShift. The Bhattacharya coefficient is shown, on the basis of which the similarity function of the histograms is determined. This indicator is displayed as a percentage of the deviation. The coordinates are displayed in time for determination the location of the object.
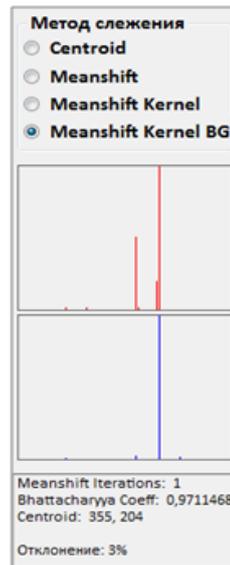
Figure 3. Graph-analytical representation of the technical vision system module operation

Conclusion

In this paper, we present an implementation of the algorithm for tracking a physical object based on the meanShift mean shift procedure. Using the technology of image pixel filtering based on color characteristics, a very accurate tracking of the moving object in real time has been obtained.

The study has been conducted with the financial support of the Science Committee of RK MES in the framework of the program target financing for the 2017-2019 biennium by the program 0006/PTF-17 «Production of titanium products for further use in medicine».

References

1. Baklanova, O.E., Baklanov, A., Shvets, O.Y.: Methods and algorithms of computer vision for automated processing of mineral rocks images. In: 10th IEEE International Symposium on Applied Computational Intelligence and Informatics, Timisoara (2015)
2. Shapiro, L., Stockman, G.: Video tracking: Theory and Practice, Wiley (2011)
3. Maggio, E., Cavallaro, A.: Computer Vision (2000)
4. Barinova, O., Veznevets, A.: Methods the image segmentation: automatic segmentation, http://cgm.computergraphics.ru/content/view/172
5. Zhang, Y.: Advances in Image And Video Segmentation, USA: IRM Press (2006)
6. Collins, R.: Mean-shift Blob Tracking through Scale Space. In: IEEE Computer Vision and Pattern Recognition, pp. 234-240. IEEE Press, Madison (2003)

7. Comaniciu, D., Ramesh, V., and Meer, P.: The Variable Bandwidth Mean Shift and Data-Driven Scale Selection: Proc Eighth Int'l Conf. Computer Vision, pp. 438-445 (2001)
8. Szeliski, R., Computer Vision: Algorithms and Applications. Microsoft Research, http://szeliski.org/Book
9. Comaniciu, D. and Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24., pp. 603-619 (2002)
10. Birchfield, S., Rangarajan, S. Spatiograms Versus Histograms for Region-Based Tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 1158-1163, San Diego, California (2005)
11. Vrazhnov, D., Shapovalov, A., Nikolaev, V.: On the quality of the work of tracking algorithms for objects on video. Computer studies and modeling. 4(2), 303-313 (2012)
12. Dimashova, M,P. Implementation of the algorithm for segmentation of images MeanShift on the GPU, http://docplayer.ru/
13. Smeulders, A., Chu, D., Cucchiara, R., Calderara, S., Dehgghan, A., Shah, M.: Visual tracking: an experimental survey. In: IEEE Transactions on Pattern Analysis and Mashine Intelligence, 36(7), pp.1442-1468 (2014)
14. Wangsheng, Y., Zhiqiang. H., Xiaohua, T., and Dan. H.: Robust Mean Shift Tracking Based on Refined Appearance Model and Online Update: Computer Vision: CCF Chinese Conference, part 1, pp.114-123 (2015).
15. Rao, S., de Medeiros Martins, A., Principe, J.C.: Mean Shift: An Information Theoretic Perspective. Pattern Recogn. Lett. 30(3), pp.222-230 (2009)
16. Comaniciu, D., Ramesh, V., Meer P.: Kernel-based object tracking/ In.: IEEE Transactions on pattern analysis and machine intelligence, Vol. 25(5) (2003)
17. Petrovsky, I.: Lectures on the theory of ordinary differential equations. MSU Publishing House (1984)
18. Bhattacharyya, A.: On a measure of divergence between two statistical populations defined by their probability distributions: Bulletin of the Calcutta Mathematical Society, pp. 99–109 (1943)
19. Gonzalez, R., Woods, R.: Digital image processing. Technosphere (2005)
20. Comaniciu, D., Ramesh V., and Meer, P.: Real-Time Tracking of Non-Rigid Objects Using Mean Shift: Proc. Conf. Computer Vision and Pattern Recognition, Vol. 2, pp. 142-149 (2000)

Study of a difference scheme for a model three-phase non-isothermal flow problem

D. Baigereyev[1], Zh. Rakhmetullina[1], R. Mukhamedova[1], D. Omariyeva[1], and R. Mukasheva[2]

[1] D. Serikbayev East Kazakhstan State Technical University,
Department of Mathematical and Computer Modeling,
19, Serikbayeva, 070004 Ust-Kamenogorsk, Kazakhstan
[2] Nazarbayev Intellectual Schools,
53, Satpayeva, 070000 Ust-Kamenogorsk, Kazakhstan
dbaigereyev@gmail.com

Abstract The present paper focuses on numerical implementation of a model describing three-phase non-isothermal flows in porous media. A finite difference scheme is proposed for numerical solution of the problem. Using the method of energy inequalities, an a priori estimate is obtained that proves the convergence of the approximate solution to the solution of the original differential problem. The proof of the main result is based on seven preliminary lemmas. An algorithm for the numerical solution based on Newton's iterative method is proposed. Numerical experiments were carried out to examine the efficiency of the proposed algorithm.

Keywords: three-phase non-isothermal flow, global pressure, finite difference method, convergence, iterative method, numerical simulation

Introduction

Modeling of three-phase non-isothermal flows in porous media is of great importance in connection with its use in the oil industry, in particular, in the development of heavy oil deposits. This is due to the fact that currently the world's heavy oil reserves are several times higher than the reserves of medium and light oils. Despite the large number of studies dedicated to both theoretical and applied aspects of three-phase non-isothermal flows in porous media, for example [1,2], many issues remain unresolved. One of such problems is a rigorous theoretical study related to numerical implementation of the model describing this process.

The governing equations of the model include the mass and energy conservation equations, Darcy's law, the equation of state, relations for capillary pressures, and the saturation constraint equation. The most common approach to the numerical solution of this problem is based on the selection of pressure of one of the phases, temperature and saturations as the primary variables. However, this approach leads to a number of difficulties arising in the numerical solution of the problem. These difficulties are mainly related to the unbounded growth of the capillary pressure gradients at residual saturations [3].

In [4], a new formulation of the two-phase isothermal flow problem was proposed which is based on the introduction of a new variable, global pressure. As a result, the capillary pressure gradients do not appear explicitly in the equation set. Despite the lack of a clear physical meaning of the global pressure, its use made it possible to obtain important theoretical results such as the proof of the existence of a solution in a two-phase case. In [5], this approach was generalized to the three-phase non-isothermal case under some simplifying assumptions about physical data.

The main difficulty in the numerical solution of this problem is connected with the complexity and strong non-linearity of the equations. Therefore, the development of computational algorithms for the numerical implementation of this problem, which require a minimum of computational operations, becomes relevant.

In authors' previous study [6] the stability of a finite difference scheme for this problem was proved using the method of energy inequalities. Three cost-effective difference schemes were constructed on the base of the studied scheme. The efficiency of the proposed algorithms was analyzed on the basis of comparing the average time spent on the numerical implementation of one time layer.

In this paper, the authors continue the research in this direction aiming at studying numerical methods for solving three-phase non-isothermal flow problems. The main difference from the problem considered in [6] is the explicit dependence of the coefficient of time derivative on temperature. Thus, the term containing the time derivative is nonlinear. In addition, capillary effects are taken into account.

Using the method of energy inequalities, the convergence of the approximate solution to the solution of the differential problem is proved. The proof of the convergence theorem is mainly based on results proposed in [6]. The resulting equations are solved using Newton's iteration method. The analysis of computational experiments is conducted in the fifth section of the paper.

Formulation of the problem

The paper focuses on studying a three-phase non-isothermal flow problem in $Q = \overline{\Omega} \times [0, t_1]$, $\overline{\Omega} = [0, 1]$ considered in [5]. It is assumed that there is an injection well inside the domain, and production wells are placed at the ends of the segment $\overline{\Omega}$.

This process is described by the following set of equations [5]:

$$\frac{\partial}{\partial t} \Phi(T) + u \frac{\partial T}{\partial x} - \frac{\partial}{\partial x} \left(k_h \frac{\partial T}{\partial x} \right) = f_T, \tag{1}$$

$$\beta_p \frac{\partial p}{\partial t} - \frac{\partial}{\partial x} \left(k_p \frac{\partial p}{\partial x} \right) = \beta_T \frac{\partial T}{\partial t} + f_p, \tag{2}$$

$$\frac{\partial s_w}{\partial t} - \frac{\partial}{\partial x} \left(a_w \frac{\partial s_w}{\partial x} \right) - \frac{\partial}{\partial x} \left(\nu_w \frac{\partial}{\partial x} (p + p_c) \right) = f_w, \tag{3}$$

$$\frac{\partial s_g}{\partial t} - \frac{\partial}{\partial x}\left(a_g \frac{\partial s_g}{\partial x}\right) - \frac{\partial}{\partial x}\left(\nu_g \frac{\partial}{\partial x}\left(p + p_c\right)\right) = f_g, \tag{4}$$

$$s_o = 1 - s_w - s_g, \tag{5}$$

$$u = -k\lambda \frac{\partial p}{\partial x} \tag{6}$$

where subscripts w, o, g denote the phases of water, oil, and steam, respectively; k is the absolute permeability, T is temperature, p is pressure, u is velocity, and s_α is the saturation of the phase α;

$$\Phi\left(T\right) = \int_0^T c_T\left(\xi\right) d\xi$$

where c_T is a function of temperature; p_c is a global capillary pressure [5] for which the following relation holds:

$$\frac{\partial p_c}{\partial x} = b_1 \frac{\partial s_w}{\partial x} + b_2 \frac{\partial s_g}{\partial x} + b_3 \frac{\partial p}{\partial x} + b_4 \frac{\partial T}{\partial x}; \tag{7}$$

k_p, k_h, ν_α, b_i, β_p and β_T are some functions of $x \in \overline{\Omega}$ and time t.

The equations (1)-(6) are complemented with initial and boundary conditions:

$$T = T_0, \;\; p = p_0, \;\; s_\alpha = s_{\alpha 0}, \;\; x \in \overline{\Omega}, \; t = 0, \tag{8}$$

$$-k_h \frac{\partial T}{\partial x} = 0, \;\; p = p_0, \;\; -a_\alpha \frac{\partial s_\alpha}{\partial x} = 0, \;\; \alpha = w, g, \; x = 0, \; x = 1, \; t > 0. \tag{9}$$

The following assumptions concerning the coefficients in (1)-(6) are made. Suppose that the following conditions hold for k_p, k_h, ν_α, and b_i:

$$k_p\left(x, t\right) \geq c_0 > 0, \;\; \left|\frac{\partial k_p}{\partial x}\right| \leq c_1, \tag{10}$$

$$k_h \geq c_0, \tag{11}$$

$$\nu_\alpha \leq c_1, \;\; \alpha = w, g, \tag{12}$$

$$\left(\left|b_1\right|, \left|b_2\right|\right) \geq c_6 > 0, \;\; \left(\left|b_3\right|, \left|b_4\right|\right) \leq c_7, \;\; c_7 > 0, \tag{13}$$

and the function ν_α can take the zero value at some points $(x, t) \in Q$. Assume that λ is a constant ($c_0 \leq \lambda \leq c_1$).

$p_{ow}\left(s_w\right)$ and $p_{go}\left(s_g\right)$ are capillary pressures which satisfy the following conditions:

$$\left|p'_{ow}\left(s_w\right)\right| \geq c_0, \;\; \left|p'_{go}\left(s_g\right)\right| \geq c_0. \tag{14}$$

We make an additional assumption that the function $\Phi\left(T\right)$ has the first and second bounded derivatives:

$$\left|\Phi'\left(T\right)\right| \leq c_5, \;\; 0 < c_4 \leq \left|\Phi''\left(T\right)\right| \leq c_5. \tag{15}$$

Formulation of the difference problem

The finite difference method is used for the numerical solution of the problem (1)-(9) in this paper. Let us introduce a uniform grid $\overline{\Omega}_{h\tau}$ in Q with steps h and τ in space and time, respectively. Associate the following difference problem with the differential problem (1)-(9) in $\overline{\Omega}_{h\tau}$:

$$\left(\Phi\left(T^h\right)\right)_t + L\left(\hat{u}^h, \hat{T}^h\right) + \Lambda_1 \hat{T}^h = f_T^h, \tag{16}$$

$$\beta_p^h p_t^h + \Lambda_2 \hat{p}^h = \beta_T^h T_t^h + f_p^h, \tag{17}$$

$$s_{\alpha,t}^h + \Lambda_{3\alpha}\hat{p}^h + \Lambda_{3\alpha}p_c^h + \Lambda_{4\alpha}\hat{s}_\alpha^h = f_\alpha^h, \quad \alpha = w, g, \tag{18}$$

$$u^h = -k\lambda p_{\bar{x}}^h, \tag{19}$$

$$T^h(x,0) = T_0, \quad p^h(x,0) = p_0, \quad s_\alpha^h(x,0) = s_{\alpha 0}, \quad x \in \overline{\Omega}_{h\tau} \tag{20}$$

where

$$L(v,\,w) = 0.5\left(\beta^+(x)\,v^{+1}w_x + \beta^-(x)\,vw_{\bar{x}}\right),$$

$$\beta^+(x) = \{2,\, x=0;\ 1,\, x \in \Omega_h;\ 0,\, x=1\}, \quad \beta^-(x) = 2 - \beta^+(x),$$

$$\Lambda_i w = \left\{-2h^{-1}\breve{y}^{+1}w_x,\ x=0;\ -\left(\breve{y}w_{\bar{x}}\right)_x,\ x \in \Omega_h;\ 2h^{-1}\breve{y}w_{\bar{x}},\ x=1\right\},$$

where $y = k_h$ for $i = 1$; $y = \nu_\alpha$ for $i = 3\alpha$, and $y = a_\alpha$ for $i = 4\alpha$; $\Lambda_2 w = -\left(\breve{k}_p w_{\bar{x}}\right)_x$. The following scalar products and norms are used hereinafter:

$$(w,\,\tilde{w})_{\overline{\Omega}_h} = \sum_{\overline{\Omega}_h} w(x)\,\tilde{w}(x)\,h, \tag{21}$$

$$\|w\|_0^2 = (w,\,w)_{\overline{\Omega}_h}, \quad \|\nabla w\|_0^2 = \sum_{\Omega_h^+} w_{\bar{x}}^2 h, \quad \|w\|_1^2 = \|\nabla w\|_0^2 + \|w\|_0^2,$$

$$\|w\|_2^2 = \sum_{\Omega_h} w_{\bar{x}x}^2 h, \quad \|y\|_3^2 = \left(\nu_w,\, y_{\bar{x}}^2\right), \quad \|y\|_4^2 = \left(\nu_g,\, y_{\bar{x}}^2\right),$$

$$\|y\|_6^2 = \|y\|_0^2 + \|y\|_3^2, \quad \|y\|_7^2 = \|y\|_0^2 + \|y\|_4^2,$$

$$\|\varphi\|_{-1} = \sup_{w \neq 0} \frac{\left|(\varphi,\,w)_{\overline{\Omega}_h}\right|}{\|w\|_1}, \quad \|w\|_C = \max_{x \in \overline{\Omega}_h}|w(x)|.$$

Suppose that the following inequality holds for initial values of pressure and temperature:

$$c_8 - q_0 - \left\|T^0\right\|_0^2 - \left\|p^0\right\|_0^2 \geq 0, \quad c_8 > 0 \tag{22}$$

where $q_0 = c_1^2 c_4 W_n\left(p_{inj}^2 + T_{inj}^2\right)$, W_n is the number of wells, p_{inj}, T_{inj} are pressure and discharge temperature of the agent, respectively. In addition, we assume that the following conditions hold:

$$\beta_p \leq c_2\tau, \quad \beta_T \leq c_2\tau, \quad c_2 > 0, \tag{23}$$

$$k \leq c_3\tau, \quad c_3 > 0. \tag{24}$$

The remaining notations are defined in [6,7].

Study of the convergence of the difference scheme

Let T, p, s_w, s_g, u be the solution of the differential problem (1)-(9), and T^h, p^h, s_w^h, s_g^h, u^h be the solution of the difference problem (16)-(20). Let us consider the following problem for $\theta = T^h - T$, $\pi = p^h - p$, $\sigma_w = s_w^h - s_w$, $\sigma_g = s_g^h - s_g$, and $\zeta = u^h - u$:

$$\left(\Phi\left(T^h\right)\right)_t - \left(\Phi\left(T\right)\right)_t + L\left(\hat{u}^h, \hat{T}^h\right) - L\left(\hat{u}, \hat{T}\right) + \Lambda_1\hat{\theta} = f_T - L\left(\hat{u}, \hat{T}\right) - \Lambda_1\hat{T}, \quad (25)$$

$$\beta_p^h \pi_t + \Lambda_2\hat{\pi} = \beta_T^h \theta_t + \psi_p, \quad (26)$$

$$\sigma_{\alpha,t} + \Lambda_{3\alpha}\hat{\pi} + \Lambda_{3\alpha}\pi_c + \Lambda_{4\alpha}\hat{\sigma}_\alpha = \psi_\alpha, \quad (27)$$

$$\zeta = -k\lambda\pi_{\bar{x}}, \quad (28)$$

$$\theta\left(0\right) = 0, \quad \pi\left(0\right) = 0, \quad \sigma_\alpha\left(0\right) = 0, \quad \alpha = w, g. \quad (29)$$

Using the Taylor expansion, the equation (25) can be reduced to the form

$$\left(\Phi'\left(T\right) + \theta\Phi''\left(T + \chi_3\theta\right)\right)\theta_t + T_t\Phi''\left(T + \chi_3\theta\right)\cdot\theta + \tau T_t\Phi''\left(T^h + \chi_1\tau T_t^h\right)\cdot\theta_t +$$

$$+\tau\left(\theta_t\right)^2\Phi''\left(T^h + \chi_1\tau T_t^h\right) + L\left(\hat{u}^h, \hat{T}^h\right) - L\left(\hat{u}, \hat{T}\right) + \Lambda_1\hat{\theta} = \psi_T \quad (30)$$

where

$$\psi_T = -\frac{\tau}{2}\left(T_t\right)^2\left(\Phi''\left(T^h + \chi_1\tau T_t^h\right) - \Phi''\left(T + \chi_2\tau T_t\right)\right) - L\left(\hat{u}, \hat{T}\right) - \Lambda_1\hat{T} + f_T.$$

We assume that the solution of the problem (16)-(20) exists, is unique and bounded. Let us first formulate the main result of this chapter.

Theorem 1. The solution of the difference problem (16)-(20) converges to the solution of the differential problem (1)-(9) under the conditions (7), (10)-(12), (14), (15), (22)-(24), and the following inequality holds:

$$d_{43}\left\|\hat{\theta}\right\|_5^2 + d_{43}\left\|\hat{\pi}\right\|_5^2 + d_{44}\left\|\hat{\sigma}_w\right\|_6^2 + d_{44}\left\|\hat{\sigma}_g\right\|_7^2 \leq$$

$$\leq d_{45}\left\|\theta\right\|_5^2 + d_{45}\left\|\pi\right\|_5^2 + d_{46}\left\|\sigma_w\right\|_6^2 + d_{46}\left\|\sigma_g\right\|_6^2 +$$

$$+d_{21}\tau\left\|\psi_T\right\|_{-1}^2 + d_{47}\left(\tau\right)\left\|\psi_p\right\|_{-1}^2 + d_{32}\left(\tau\right)\left\|\psi_w\right\|_{-1}^2 + d_{42}\left(\tau\right)\left\|\psi_g\right\|_{-1}^2$$

where d_i are positive constants and $d_k(\tau)$ are positive-definite functions which will be defined below.

Before proving the theorem, we give the following auxiliary lemmas. Let us formulate without proof the following lemma.

Lemma 1. The following estimates hold under the conditions (10), (12):

$$\left(\Lambda_\alpha v, v\right) \geq c_0\left\|\nabla v\right\|_0^2, \quad \alpha = 1, 2, \quad (31)$$

$$\left(\Lambda_{3\alpha} w, \tilde{w}\right) \leq \frac{1}{2}c_1 M\varepsilon\left\|\nabla w\right\|_0^2 + \frac{c_1}{2\varepsilon}\left\|\nabla\tilde{w}\right\|_0^2, \quad \varepsilon > 0. \quad (32)$$

Lemma 2. The following estimates hold under the conditions (10), (23):

$$c_0 \left\| \hat{p}^h \right\|_0^2 + c_0 \tau^2 \left\| p_t^h \right\|_0^2 + d_1 \tau \left\| \nabla \hat{p}^h \right\|_0^2 \le c_0 \left\| p^h \right\|_0^2 + \frac{c_2 \varepsilon \tau^3}{2} \left\| T_t^h \right\|_0^2 + \frac{\tau \varepsilon}{2} \left\| f_p^h \right\|_0^2, \quad (33)$$

$$d_2 \tau \left\| \hat{p}^h \right\|_2^2 \le \left\| p^h \right\|_0^2 + c_2 \varepsilon \tau^3 \left\| T_t^h \right\|_0^2 + \tau \varepsilon \left\| f_p^h \right\|_0^2, \quad \varepsilon > 0. \quad (34)$$

Proof. The inequality (33) is simply obtained by multiplying the equation by $2\tau \hat{p}$ and using the conditions (10), (23), where the constant d_1 is defined as $d_1 = 2c_0 - \varepsilon^{-1} (c_2 + 1)$. To obtain the inequality (34), multiply the equation (17) by $-\tau \hat{p}_{\bar{x}x}$ and use the conditions (10), (23):

$$d_2 \tau \left\| \hat{p}^h \right\|_2^2 \le c_1 \varepsilon \tau (0.5 + c_1) \left\| \nabla \hat{p}^h \right\|_0^2 + \frac{c_2 \varepsilon \tau^3}{2} \left\| T_t^h \right\|_0^2 + \frac{\tau \varepsilon}{2} \left\| f_p^h \right\|_0^2 \quad (35)$$

where $d_2 = c_0 - (2\varepsilon)^{-1} (c_2 + 1)$. Add the inequalities (33) and (35) to obtain

$$c_0 \left\| \hat{p}^h \right\|_0^2 + c_0 \tau^2 \left\| p_t^h \right\|_0^2 + \tau d_3 \left\| \nabla \hat{p}^h \right\|_0^2 + d_2 \tau \left\| \hat{p}^h \right\|_2^2 \le$$

$$\le c_0 \left\| p^h \right\|_0^2 + c_2 \varepsilon \tau^3 \left\| T_t^h \right\|_0^2 + \tau \varepsilon \left\| f_p^h \right\|_0^2$$

where $d_3 = 2c_0 - \varepsilon^{-1} (c_2 + 1) - \varepsilon \left(0.5c_1 + c_1^2 \right)$. Choosing ε from conditions $d_1 > 0$, $d_2 > 0$, $d_3 > 0$, we obtain the inequality (34).

To estimate the convective term in the equation for temperature, we use the approach used in [6]. The analog of Lemma 4 in [6] is the following lemma.

Lemma 3. The following estimate holds under the conditions (22), (24), (11):

$$c_2 \left\| \hat{T}^h \right\|_0^2 + d_6 \tau^2 \left\| T_t^h \right\|_0^2 + d_{11} \tau \left\| \nabla \hat{T}^h \right\|_0^2 + d_5 c_0 \left\| \hat{p}^h \right\|_0^2 +$$

$$+ d_9 \tau \left\| \nabla \hat{p}^h \right\|_0^2 + d_5 c_0 \tau^2 \left\| p_t^h \right\|_0^2 \le c_8.$$

Proof. Carrying out calculations analogous to those made in deriving the equation (30), and neglecting the term with first order of smallness relative to τ, we obtain the following equation for T^h:

$$T_t^h \cdot \Phi' \left(T^h \right) + L \left(\hat{u}^h, \hat{T}^h \right) + \Lambda_1 \hat{T}^h = f_T^h. \quad (36)$$

Multiply this equation by $2\tau \hat{T}^h$:

$$c_2 \left\| \hat{T}^h \right\|_0^2 - c_2 \left\| T^h \right\|_0^2 + c_2 \tau^2 \left\| T_t^h \right\|_0^2 + 2\tau \left(L \left(\hat{u}^h, \hat{T}^h \right), \hat{T}^h \right) +$$

$$+ 2\tau \left(\Lambda_1 \hat{T}^h, \hat{T}^h \right) = 2\tau \left(f_T^h, \hat{T}^h \right). \quad (37)$$

Estimate the scalar products in (37):

$$2\tau \left(L \left(\hat{u}^h, \hat{T}^h \right), \hat{T}^h \right) \le 2\tau \left| \sum_{\Omega_h^-} \hat{u}^{h,+1} \hat{T}_x^h \hat{T}^h + \sum_{\Omega_h^+} \hat{u}^h \hat{T}_{\bar{x}}^h \hat{T}^h \right| h \le$$

$$\leq 4\tau \left\|\nabla \hat{u}^h\right\|_0 \left\|\hat{T}^h\right\|_0 \left|\sum_{\Omega_h^-} \hat{T}_x^h + \sum_{\Omega_h^+} \hat{T}_{\bar{x}}^h\right| h \leq$$

$$\leq d_2\tau \left\|\hat{p}^h\right\|_2 \left\|\hat{T}^h\right\|_0 \left\|\nabla \hat{T}^h\right\|_0 \leq \frac{\varepsilon_1 d_3 \tau}{2}\left\|\hat{p}^h\right\|_2^2 + \frac{d_3 \tau}{\varepsilon_2}\left\|\hat{T}^h\right\|_0^2\left\|\nabla \hat{T}^h\right\|_0^2.$$

Using Lemma 2 and the difference analogue of the embedding theorem, estimate the right-hand side of the last inequality as follows:

$$2\tau \left(L\left(\hat{u}^h, \hat{T}^h\right), \hat{T}^h\right) \leq d_4 \left\|\nabla p^h\right\|_0^2 + d_4\varepsilon_3\tau^3 \left\|T_t^h\right\|_0^2 +$$

$$+d_4\tau\varepsilon_3 \left\|f_p^h\right\|_0^2 + \frac{d_3\tau}{\varepsilon_2}\left\|\hat{T}^h\right\|_0^2\left\|\nabla \hat{T}^h\right\|_0^2.$$

The right-hand side of (37) is estimated as follows:

$$2\tau \left(f_T^h, \hat{T}^h\right) \leq \varepsilon_2\tau \left\|f_T^h\right\|_0^2 + \frac{d_{16}\tau}{\varepsilon_2}\left\|\nabla \hat{T}^h\right\|_0^2.$$

Use these inequalities to estimate the terms in (37), and add this inequality to the inequality (33) multiplied by some $d_5 > 0$ to obtain

$$c_2 \left\|\hat{T}^h\right\|_0^2 + \tau^2\left(c_2 - d_4\varepsilon_3\tau - 0.5d_5c_2\varepsilon_4\tau\right)\left\|T_t^h\right\|_0^2 +$$

$$+\tau\left(2c_0 - \frac{d_{16}}{\varepsilon_2} - \frac{d_3}{2\varepsilon_2}\left\|\hat{T}^h\right\|_0^2\right)\left\|\nabla \hat{T}^h\right\|_0^2 + d_5c_0\left\|\hat{p}^h\right\|_0^2 +$$

$$+d_5c_0\tau^2\left\|p_t^h\right\|_0^2 + \tau\left(d_1d_5 - d_4\right)\left\|\nabla \hat{p}^h\right\|_0^2 \leq$$

$$\leq c_2 \left\|T^h\right\|_0^2 + d_5c_0\left\|p^h\right\|_0^2 + \varepsilon_2\tau \left\|f_T^h\right\|_0^2 + \tau\left(\frac{d_5\varepsilon_4}{2} + d_4\varepsilon_3\right)\left\|f_p^h\right\|_0^2.$$

Choosing $\tau < \tau_0$, where $\tau_0 = c_2\left(d_4\varepsilon_3 - 0.5d_5c_2\varepsilon_4\right)^{-1}$, the last inequality yields

$$c_2 \left\|\hat{T}^h\right\|_0^2 + d_6\tau^2 \left\|T_t^h\right\|_0^2 + d_7\tau\left(c_8 - \left\|\hat{T}^h\right\|_0^2\right)\left\|\nabla \hat{T}^h\right\|_0^2 + d_5c_0\left\|\hat{p}^h\right\|_0^2 +$$

$$+d_5c_0\tau^2\left\|p_t^h\right\|_0^2 + d_9\tau\left\|\nabla \hat{p}^h\right\|_0^2 \leq c_2 \left\|T^h\right\|_0^2 + d_5c_0\left\|p^h\right\|_0^2 + \varepsilon_2\tau \left\|f_T^h\right\|_0^2 + d_{10}\tau\left\|f_p^h\right\|_0^2$$

where $d_7 = (2d_{16} + d_3)(2\varepsilon_2)^{-1}$, $c_8 = 4c_0\varepsilon_2(2d_{16} + d_3)^{-1}$, $d_9 = d_1d_5 - d_4$, $d_{10} = 0.5d_5\varepsilon_4 + d_4\varepsilon_3$. Choose ε_i and d_5 from the conditions of non-negativity of the coefficients under the norms on the left-hand side of the inequality. Applying the induction method as in Lemma 4 of the work [6], we obtain the assertion of the lemma.

Lemma 4. The following estimate holds under the conditions (11), (24):

$$\left(L\left(\hat{u}^h, \hat{T}^h\right) - L\left(\hat{u}, \hat{T}\right), \hat{\theta}\right) \leq \frac{3\varepsilon\tau d_{11}}{2}\left\|\hat{p}^h\right\|_2^2 +$$

$$+\frac{3\tau d_{11}}{2\varepsilon}\left\|\hat{T}^h\right\|_1^2\left\|\nabla\hat{\theta}\right\|_0^2+\frac{\tau}{\varepsilon}\left\|\nabla\hat{\theta}\right\|_0^2,\ \ \varepsilon>0.$$

Proof. Using the definition of L, the summation by parts, the ε-inequality and the condition (24), we obtain:

$$\left(L\left(\hat{u}^h,\hat{T}^h\right)-L\left(\hat{u},\hat{T}\right),\hat{\theta}\right)\le\left\|\nabla\hat{u}^k\right\|_0\left|\sum_{\Omega_h^+}\hat{T}^h\hat{\theta}_{\bar{x}}+\sum_{\Omega_h^-}\hat{T}^h\hat{\theta}_x\right|h+$$

$$+2\left\|\nabla\hat{u}^h\right\|_0\left|\hat{T}_0^h\hat{\theta}_0-\hat{T}_N\hat{\theta}_N\right|+2\tau c_9\left\|\nabla\hat{u}\right\|_0\left\|\nabla\hat{\theta}\right\|_0\le$$

$$\le d_{11}\tau\left\|\hat{p}^h\right\|_2\left\|\hat{T}^h\right\|_0\left\|\nabla\hat{\theta}\right\|_0+2\tau d_{11}\left\|\hat{p}^h\right\|_2\left\|\nabla\hat{T}^h\right\|_0\left\|\nabla\hat{\theta}\right\|_0+2\tau c_9\left\|\nabla\hat{u}\right\|_0\left\|\nabla\hat{\theta}\right\|_0\le$$

$$\le\frac{\varepsilon\tau d_{11}}{2}\left\|\hat{p}^h\right\|_2^2+\frac{\tau d_{11}}{2\varepsilon}\left\|\hat{T}^h\right\|_0^2\left\|\nabla\hat{\theta}\right\|_0^2+\varepsilon\tau d_{11}\left\|\hat{p}^h\right\|_2^2+$$

$$+\frac{\tau d_{11}}{\varepsilon}\left\|\nabla\hat{T}^h\right\|_0^2\left\|\nabla\hat{\theta}\right\|_0^2+2\tau\varepsilon c_9\left\|\nabla\hat{u}\right\|_0^2+\frac{\tau}{\varepsilon}\left\|\nabla\hat{\theta}\right\|_0^2.$$

Using the definition of the norm $\|\cdot\|_1$ and taking into account the assumption of boundedness of $\|\nabla\hat{u}\|^2$, we obtain the assertion of the lemma.

Lemma 5. The following estimate holds under the conditions (11), (15), (22):

$$c_4c_8\left\|\hat{\theta}\right\|_0^2+d_{18}\left\|\nabla\hat{\theta}\right\|_0^2\le d_{19}\left\|\theta\right\|_0^2+2d_{12}\varepsilon_2\left\|p^h\right\|_0^2+\varepsilon_1\left\|\psi_T\right\|_{-1}^2.$$

Proof. Reduce the equation (30) to the form

$$-\left|\Phi'\left(T\right)\right|\cdot\theta_t-\tau T_t\left|\Phi''\left(T^h+\chi_1\tau T_t^h\right)\right|\cdot\theta_t-\left|\Phi''\left(T+\chi_3\theta\right)\right|\cdot\theta\theta_t-\left|\Phi''\left(T+\chi_3\theta\right)\right|\times$$

$$\times T_t\theta-\tau\left|\Phi''\left(T^h+\chi_1\tau T_t^h\right)\right|\left(\theta_t\right)^2+L\left(\hat{u}^h,\hat{T}^h\right)-L\left(\hat{u},\hat{T}\right)+\Lambda_1\hat{\theta}\le\psi_T.$$

Multiply this equation by $-2\hat{\theta}$:

$$2\left(\left|\Phi'\left(T\right)\right|\cdot\theta_t,\hat{\theta}\right)+2\tau\left(T_t\left|\Phi''\left(T^h+\chi_1\tau T_t^h\right)\right|\cdot\theta_t,\hat{\theta}\right)+$$

$$+\left(\left|\Phi''\left(T+\chi_3\theta\right)\right|\cdot\theta\theta_t,\hat{\theta}\right)+\left(\left|\Phi''\left(T+\chi_3\theta\right)\right|T_t\theta,\hat{\theta}\right)+$$

$$+\left(\tau\left|\Phi''\left(T^h+\chi_1\tau T_t^h\right)\right|\left(\theta_t\right)^2,2\hat{\theta}\right)-$$

$$-2\left(L\left(\hat{u}^h,\hat{T}^h\right)-L\left(\hat{u},\hat{T}\right),\hat{\theta}\right)-2\left(\psi_T,\hat{\theta}\right)\le0. \qquad (38)$$

Estimating scalar products in (38) using the conditions (11), (15), (22), Lemma 2 and Lemma 3, it is easy to obtain the inequality

$$c_4c_8\left\|\hat{\theta}\right\|_0^2+\tau^2\left(c_4c_8-\frac{2c_5}{\varepsilon_1}\left\|\theta_t\right\|_0^2\right)\left\|\theta_t\right\|_0^2+\left(2c_1-d_{15}-\varepsilon_1\left(1+2c_5\right)\left\|\theta_t\right\|_0^2\right)\times$$

$$\times \left\| \nabla \hat{\theta} \right\|_0^2 \leq \left(c_4 c_8 + \frac{c_5}{\varepsilon_1} + \frac{c_9 c_5}{\varepsilon_1} \right) \|\theta\|_0^2 + 2 d_{12} \varepsilon_2 \left\| p^h \right\|_0^2 + \varepsilon_1 \|\psi_T\|_{-1}^2 .$$

Assuming that the condition $\|\theta_t\|_0^2 < d_{17}$ holds, where

$$d_{17} = \min \left\{ c_4 c_8 \varepsilon_1 \left(2 c_5 \right)^{-1}, \ \left(2 c_1 - d_{15} \right) \varepsilon_1^{-1} \left(1 + 2 c_5 \right)^{-1} \right\},$$

we arrive at the assertion of the lemma.

The following lemma is proved as Lemma 3 in [6]:

Lemma 6. The following estimate holds under the conditions (10), (23):

$$\|\hat{\pi}\|_0^2 + 2 \tau c_0 \|\hat{\pi}\|_1^2 \leq d_{22} \|\pi\|_0^2 + 4 \tau^2 d_{22} \|\theta_t\|_0^2 + \tau \left(1 + \tau \right) \varepsilon_2 \|\psi_p\|_{-1}^2 .$$

To estimate the terms containing the global capillary pressure in the equations for water and steam saturations, we reduce these equations to the equivalent form using the relation (7):

$$\sigma_{w,t} + \Lambda_{3w} \hat{\pi} + \Lambda_{4w} \hat{\sigma}_w + \Lambda_{5w} \pi + \Lambda_{6w} \sigma_w + \Lambda_{7w} \sigma_g + \Lambda_{8w} \theta = \psi_w, \qquad (39)$$

where

$$\Lambda_{k\alpha} = \left\{ -\frac{2}{h} \left(\check{\nu}_\alpha^h b_i^h \right)^{+1} w_x, \ x = 0; \ - \left(\check{\nu}_\alpha^h b_i^h w_{\bar{x}} \right)_x, \ x \in \Omega_h; \ \frac{2}{h} \check{\nu}_\alpha^h b_i^h w_{\bar{x}}, \ x = 1 \right\},$$

$i = 3, 1, 2, 4$ for $\alpha = 5, 6, 7, 8$, respectively.

Lemma 7. The following estimates hold under the conditions (12), (14), (7):

$$\|\hat{\sigma}_w\|_0^2 + d_{23} \tau \|\hat{\sigma}_w\|_3^2 + d_{24} \tau \|\sigma_g\|_3^2 \leq$$

$$\leq d_{25} \|\sigma_w\|_0^2 + d_{26} \tau \varepsilon_1 \|\nabla \hat{\pi}\|_0^2 + d_{27} \tau \|\nabla \pi\|_0^2 + d_{28} \tau \|\sigma_w\|_3^2 +$$

$$+ d_{29} \tau \|\sigma_g\|_3^2 + d_{30} \tau \|\nabla \theta\|_0^2 + d_{31} \left(\tau \right) \|\sigma_g\|_0^2 + d_{32} \left(\tau \right) \|\psi_w\|_{-1}^2 ,$$

$$\|\hat{\sigma}_g\|_0^2 + d_{33} \tau \|\hat{\sigma}_g\|_4^2 + d_{34} \tau \|\sigma_w\|_4^2 \leq$$

$$\leq d_{35} \|\sigma_g\|_0^2 + d_{36} \tau \varepsilon_1 \|\nabla \hat{\pi}\|_0^2 + d_{37} \tau \|\nabla \pi\|_0^2 + d_{38} \tau \|\sigma_g\|_4^2 +$$

$$+ d_{39} \tau \|\sigma_w\|_4^2 + d_{40} \tau \|\nabla \theta\|_0^2 + d_{41} \left(\tau \right) \|\sigma_w\|_0^2 + d_{42} \left(\tau \right) \|\psi_g\|_{-1}^2 .$$

Proof. Multiply the equation (39) by $2 \tau \left(\hat{\sigma}_w + \sigma_g \right)$ and estimate the scalar products using the methods applied above. After simple transformations, we obtain the inequality

$$\|\hat{\sigma}_w\|^2 + \tau^2 \left(1 - \varepsilon_1^{-1} \right) \|\sigma_{w,t}\|^2 + \tau \left(2 c_0 - \left(c_1 + 1 + 2 c_6 + c_7 \right) \varepsilon_1^{-1} - 3 \varepsilon_1 \right) \|\hat{\sigma}_w\|_3^2 +$$

$$+ 2 c_6 \tau \|\sigma_g\|_3^2 \leq \left(1 + \varepsilon_1 + \varepsilon_1^{-1} \right) \|\sigma_w\|_0^2 + 2 \tau c_1 \varepsilon_1 \|\nabla \hat{\pi}\|_0^2 + \varepsilon_1 c_1 c_7 \tau \left(c_7 + 1 \right) \|\nabla \pi\|_0^2 +$$

$$+ \varepsilon_1 \tau \left(c_6 + c_7 \right) \|\sigma_w\|_3^2 + \tau \left(c_6 \varepsilon_1 + \left(2 + 3 c_7 \right) \varepsilon_1^{-1} \right) \|\sigma_g\|_3^2 + 2 \tau c_1 c_7 \varepsilon_1 \|\nabla \theta\|_0^2 +$$

$$+ \left(\tau + 1 \right) \varepsilon_1^{-1} \|\sigma_g\|_0^2 + \varepsilon_1 \tau \left(2 + \tau \right) \|\psi_w\|_{-1}^2 .$$

Choosing ε_1 from the condition of non-negativity of the coefficients on the left-hand side of the resulting inequality, we arrive at the assertion of the lemma.

Now let us prove the main theorem. Combining the results of Lemma 5, Lemma 6, and Lemma 7, we obtain the following inequality after obvious transformations:

$$\left\|\hat{\theta}\right\|_0^2 + \|\hat{\pi}\|_0^2 + \|\hat{\sigma}_w\|_0^2 + \|\hat{\sigma}_g\|_0^2 + \tau^2\left(c_2 - 4d_{22}\right)\|\theta_t\|_0^2 + d_{19}\tau\left\|\nabla\hat{\theta}\right\|_0^2 +$$

$$+\tau\left(2c_0 - d_{26}\varepsilon_1 - d_{36}\varepsilon_1\right)\|\nabla\hat{\pi}\|_0^2 + +d_{23}\tau\|\hat{\sigma}_w\|_3^2 + d_{33}\tau\|\hat{\sigma}_g\|_4^2 +$$

$$+\left(d_{34} - d_{39}\right)\tau\|\sigma_w\|_4^2 + \tau\left(d_{24} - d_{29}\right)\|\sigma_g\|_3^2 \leq$$

$$\leq d_{17} + d_{20}\|\theta\|_0^2 + d_{22}\|\pi\|_0^2 + \left(d_{25} + d_{41}\left(\tau\right)\right)\|\sigma_w\|_0^2 + \left(d_{31}\left(\tau\right) + d_{35}\right)\|\sigma_g\|_0^2 +$$

$$+\tau\left(d_{27} + d_{37}\right)\|\nabla\pi\|_0^2 + \tau\left(d_{30} + d_{40}\right)\|\nabla\theta\|_0^2 + d_{28}\tau\|\sigma_w\|_3^2 + d_{38}\|\sigma_g\|_4^2 +$$

$$+d_{21}\tau\|\psi_T\|_{-1}^2 + \varepsilon_2\tau\left(1 + \tau\right)\|\psi_p\|_{-1}^2 + d_{32}\left(\tau\right)\|\psi_w\|_{-1}^2 + d_{42}\left(\tau\right)\|\psi_g\|_{-1}^2.$$

Denoting

$$d_{43} = \min\left\{1, d_{19}, 2c_0 - d_{26}\varepsilon_1 - d_{36}\varepsilon_1\right\}, \quad d_{44} = \min\left\{1, d_{23}, d_{33}\right\},$$

$$d_{45} = \max\left\{1, d_{20}, d_{30} + d_{40}, d_{22}, d_{27} + d_{37}\right\},$$

$$d_{46} = \max\left\{d_{25} + d_{41}\left(\tau\right), d_{28}, d_{31}\left(\tau\right) + d_{35}, d_{38}\right\},$$

we arrive at the assertion of the main theorem.

Numerical implementation and discussion of results

To solve the equation (16), the Newton's iterative method is used:

$$\Phi\left(\hat{T}^k\right) + \Phi'\left(\hat{T}^k\right)\left(\hat{T}^{k+1} - \hat{T}^k\right) + \tau L\left(\hat{u}, \hat{T}^{k+1}\right) + \tau \Lambda_1\hat{T}^{k+1} = \Phi\left(T\right) + \tau f_T \quad (40)$$

where the index k is the iteration number. Using the definitions of the operators Λ_1, L, we reduce the equation (40) to the three-point form and find the boundary conditions:

$$\left(\frac{\tau}{h^2} + \frac{\tau\hat{u}_{i-\frac{1}{2}}}{2h}\right)\hat{T}_{i-1}^{k+1} - \left(\frac{2\tau}{h^2} + \Phi'\left(\hat{T}_i^k\right) + \frac{\tau\left(\hat{u}_{i-\frac{1}{2}} - \hat{u}_{i+\frac{1}{2}}\right)}{2h}\right)\hat{T}_i^k +$$

$$+\left(\frac{\tau}{h^2} - \frac{\tau\hat{u}_{i+\frac{1}{2}}}{2h}\right)\hat{T}_{i+1}^{k+1} = -F_{T,i}, \tag{41}$$

$$\hat{T}_0^{k+1} = \delta_1\hat{T}_1^{k+1} + \delta_2, \quad \hat{T}_N^{k+1} = \delta_3 \tag{42}$$

where

$$\delta_1 = \frac{2\tau k_{h,1/2} + \tau h \hat{u}_{1/2}}{h^2 \Phi'\left(T_0^k\right) + 2\tau k_{h,1/2} - \tau h u_{1/2}}, \quad \delta_2 = \frac{h^2 F_{T,0}}{h^2 \Phi'\left(T_0^k\right) + 2\tau k_{h,1/2} - \tau h \hat{u}_{1/2}},$$

$$\delta_3 = \frac{\tilde{\beta}_N \tau h \hat{u}_{N-\frac{1}{2}} + 2\tau k_h \tilde{\beta}_N + h^2 F_{T,N}}{h^2 \Phi'\left(T_N^k\right) + (1 - \tilde{\alpha}_N)\left(\tau h \hat{u}_{N-\frac{1}{2}} + 2\tau k_h\right)},$$

$$F_{T,i} = \Phi\left(T_i\right) + \tau f_T - \Phi\left(\hat{T}_i^k\right) + \Phi'\left(\hat{T}_i^k\right)\hat{T}_i^k,$$

$\tilde{\alpha}_i$, $\tilde{\beta}_i$ are coefficients in the Thomas algorithm. Equations for pressure and saturation also lead to a three-point view and are implemented using the Thomas algorithm.

Computational experiments were carried out with the following choice of functions $c_T\left(T\right)$:

$$c_T\left(T\right) = 0.014 \cdot \exp 0.0084\left(T - T_0\right), \tag{43}$$

$$c_T\left(T\right) = 0.0028\left(1 + 0.011 \cdot T^2\right). \tag{44}$$

The temperature distribution found on the previous time layer is chosen as the initial approximation of the iterative process. The calculations are conducted until the following condition holds:

$$\left\|\hat{T}^{k+1} - \hat{T}^k\right\|_C \leq \varepsilon_0.$$

The number of iterations in the Newton's iterative process corresponding to cases (43) and (44) are presented in Table 1 and Table 2, respectively. Overall, the iterative process converges in less than 9 iterations in Case 1 at the selected values of parameters h and τ, and less than 11 iterations in Case 2.

Table 1. The number of iterations corresponding to Case 1

h	τ	ε_0	$t = \tau$	$t = 5\tau$	$t = 10\tau$	$t = 20\tau$	$t = 50\tau$
0.01	0.001	10^{-6}	9	7	7	7	6
0.01	0.001	$5 \cdot 10^{-5}$	7	6	6	5	5
0.01	0.001	$2 \cdot 10^{-5}$	7	6	5	5	5
0.01	0.001	$5 \cdot 10^{-4}$	5	5	5	5	4

Discussion of the simulation results was carried out in [8]. In that paper, a one-dimensional test was performed on the example of the problem of steam displacement by steam. Analysis of the results of computational experiments showed that the model obtained reproduces the characteristic features of this process. The distributions of pressure and saturations obtained as a result of solving the problem (16)-(20) are similar to those obtained in [8].

Table 2. The number of iterations corresponding to Case 2

h	τ	ε_0	$t = \tau$	$t = 5\tau$	$t = 10\tau$	$t = 20\tau$	$t = 50\tau$
0.01	0.001	10^{-6}	11	9	9	8	8
0.01	0.001	$5 \cdot 10^{-5}$	9	8	8	7	7
0.01	0.001	$2 \cdot 10^{-5}$	8	8	8	7	6
0.01	0.001	$5 \cdot 10^{-4}$	7	7	7	6	5

Conclusion

This paper studied a finite difference scheme for the solution of a one-dimensional three-phase non-isothermal flow problem. Using the method of energy inequalities, an a priori estimate is obtained which proves the convergence of the approximate solution to the solution of the differential problem. The efficiency of the proposed algorithm is verified in numerical experiments carried out for two model cases. The results obtained in this paper can be used to carry out further research on numerical implementation of models describing multiphase non-isothermal flows in porous media.

References

1. Bokserman, A.A., Yakuba, S.I.: Numerical study of the process of steam injection. Izvestiya AN SSSR 4, 78–84 (1987)
2. Abdramanova, M.B.: Chislennoe modelirovanie protsessa vytesneniya nefti parom. Vestnik KazGU. Seriya matematika, mehanika, informatika 10, 3–10 (1998)
3. Chavent, G.: A fully equivalent global pressure formulation for three-phase compressible flow, 19 p. INRIA (2009)
4. Chavent, G., Jaffre, J.: Mathematical models and finite elements for reservoir simulation, 375 p. Elsevier (1986)
5. Baigereyev, D.R.: Numerical study of the process of displacement of oil by steam based on the global pressure approach. In: The 5th International Conference on Control and Optimization with Industrial applications, pp. 307–310. Baku (2015)
6. Zhumagulov, B.T., Temirbekov, N.M., Baigereyev, D.R.: Efficient difference schemes for the three-phase non-isothermal flow problem. In: AIP Conference Proceedings 1880, pp. 060001-1 – 060001-10. American Institute of Physics, College Park (2017)
7. Samarskii, A.A.: The theory of difference schemes, 786 p. CRC Press (2001)
8. Temirbekov, N.M., Baigereyev, D.R.: Modeling of three-phase non-isothermal flow in porous media using the approach of reduced pressure. In: Communications in Computer and Information Science 549, pp. 166–176. Springer (2015)

Investigation of difference schemes of thermal convection in the (Ψ, Ω) variables

N. Danayev and F. Amenova

S. Amanzholov East Kazakhstan State University, Department of Mathematics,
148, Voroshilov str., 070002 Ust-Kamenogorsk, Kazakhstan
faramen.74@mail.ru

Abstract An implicit iterative algorithm is proposed for the numerical implementation of the solution of grid equations. For the algorithm under consideration, the following problems were tested: the problem of fluid flow in a square region (the cavity problem); the problem of free convection in a closed square region when heated from above; the problem of free convection in a closed square region when heated from the side. Calculations are carried out using an iterative algorithm of the variable direction type.

Keywords: Difference equations, operator-difference equations of thermal convection, estimation of the convergence rate, iterative algorithm, fluid flow

Introduction

For a numerical solution of thermal convection of an incompressible fluid, methods based on the approximation of Navier-Stokes equations written in the *stream function, velocity vortex* variables are widely used. In the literature there are works on the justification of iterative algorithms in the case of solving grid stationary problems of thermal convection of an incompressible fluid. In the work of A. F. Voevodin [1] , the absolute stability of classical implicit difference schemes for the two-dimensional Stokes equations by the operator inequalities method is proved, and stable direct and iterative methods for solving boundary value problems are proposed. In the work of A. F. Voevodin, T. V. Yushkov [2], a numerical method based on the method of splitting by physical processes for solving initial-boundary value problems for thermal convection, written in the *stream function, vortex* variables, is proposed. To solve systems of implicit difference equations, a modification of the "two-field" calculation of the values of the stream function and the velocity vortex is used. A stability study on the linear approximation of difference schemes is carried out [3,4].

Significant results in this area of research were obtained in the works of V. P. Vabishchevich, A. N. Tikhonov, A. A. Samarskii, N. N. Yanenko, Yu. I. Shokin, P. Roach, O. M. Belotserkovsky, E. L. Tarunin, R. Temam, O. A. Ladyzhenskaya, Abdrashin, S. L. Lapko, B. G. Kuznetsov. Among the works of Kazakh scientists,

the works of Sh. Smagulov, N. T. Danaev, B. T. Zhumagulov, N. M. Temirbekov, A. Zh. Kaltaev, Sh. N. Kuttykozhaeva, B. Rysbayuly, Zh. T. Sugirbayeva, B. A. Urmashev and others can be noted.

Formulation of the Problem

We consider a two-dimensional system of stationary equations of thermal convection in the *stream function, velocity vortex* variables for an incompressible fluid in the square domain $D = \{0 \le x, y \le 1\}$ of the following form [4,5,6]:

$$\left(\Omega \frac{\partial \Psi}{\partial y} \right)_x - \left(\Omega \frac{\partial \Psi}{\partial x} \right)_y = \Delta\Omega + \mathrm{Gr}\frac{\partial \theta}{\partial x} + f\left(x, y\right), \tag{1}$$

$$\Delta\Psi = \Omega, \quad (x, y) \in D, \tag{2}$$

$$\left(\theta \frac{\partial \Psi}{\partial y} \right)_x - \left(\theta \frac{\partial \Psi}{\partial x} \right)_y = \frac{1}{\mathrm{Pr}}\Delta\theta + g\left(x, y\right) \tag{3}$$

with boundary conditions

$$\Psi = \frac{\partial \Psi}{\partial \boldsymbol{n}} = 0, \quad \theta = 0, \quad (x, y) \in \partial D, \tag{4}$$

where Δ is the two-dimensional Laplace operator, Ψ is the stream function, Ω is vorticity, θ is temperature, Gr is the Grashof number, Pr is the Prandtl number, $\boldsymbol{n}$ is the outer normal to the boundary of the domain, $f\left(x, y\right)$, $g\left(x, y\right)$ are some given functions.

Consider a scheme on a symmetric stencil of the following form to approximate equations (1)-(3) in a finite difference domain

$$D_h = \left\{ (kh_1, mh_2), \ k = \overline{1, \, N_1 - 1}, \ m = \overline{1, \, N_2 - 1} \right\},$$

where h_1 and h_2 are grid steps in the x and y directions, respectively:

$$L_{h,\Psi}\left(\Omega\right)\Psi = \Delta_h\Omega + \mathrm{Gr}\theta_{\dot{x}} + f, \tag{5}$$

$$\Delta_h\Psi = \Omega, \tag{6}$$

$$L_{h,\theta}\left(\Psi\right)\theta = \frac{1}{\mathrm{Pr}}\Delta_h\theta + g \tag{7}$$

with boundary conditions

$$\Psi_{0m} = \Psi_{N_1 m} = \Psi_{k0} = \Psi_{kN_2} = 0, \quad \theta_{0m} = \theta_{N_1 m} = \theta_{k0} = \theta_{kN_2} = 0,$$

$$\Omega_{0m} = \frac{2}{h_1}\Psi_{x,0m}, \quad \Omega_{N_1 m} = -\frac{2}{h_1}\Psi_{\bar{x},N_1 m}, \quad m = \overline{1, \, N_2 - 1}, \tag{8}$$

$$\Omega_{k0} = \frac{2}{h_2}\Psi_{y,k0}, \quad \Omega_{kN_2} = -\frac{2}{h_2}\Psi_{\bar{y},kN_2}, \quad k = \overline{1, \, N_1 - 1}.$$

The boundary conditions for the velocity vortex are taken in the form of Thom's formulas [6]. The difference operators $L_{h,\Psi}$ and $L_{h,\theta}$ correspond to the approximation of the convective terms of the equations (1) and (3).

Suppose that the following relations are valid for the difference operators $L_{h,\Psi}$, $L_{h,\theta}$ [7]:

$$|(L_{h,\Psi}(\Omega)\Psi,\, V)| \leq c_0 \, \|\Omega\| \, \|\Delta_h\Psi\| \, \|\Delta_h V\|,$$

$$|(L_{h,\theta}(\Psi)\theta,\, V)| \leq c_0 \, \|\Delta_h\Psi\| \, \|\nabla_h\theta\| \, \|\nabla_h V\|, \tag{9}$$

$$(L_{h,\Psi}(\Omega)\Psi,\Psi) = 0, \quad (L_{h,\theta}(\Psi)\theta,\theta) = 0, \quad \forall \Psi, V, \theta \in \mathring{\Omega}_h(D_h), \tag{10}$$

where $c_0 > 0$ is a uniformly bounded constant, $\mathring{\Omega}_h(D_h)$ is the space of grid functions with zero boundary values defined on the grid D_h. We will also use the well-known inequalities [8]:

$$\delta_0 \, \|u\|^2 \leq \|\nabla_h u\|^2, \quad \delta_0 \, \|u\| \leq \|\Delta_h u\|, \quad \|\Delta_h u\|^2 \leq \frac{8}{h^2} \|\nabla_h u\|^2,$$

which are valid for any grid function $u \in \mathring{\Omega}_h(D_h)$, where $h = \min(h_1, h_2)$, $\delta_0 > 0$ is the minimal eigenvalue of the Laplace difference operator.

The difference operators $L_{h,\Psi}$, $L_{h,\theta}$ satisfying the conditions (9), (10), can be chosen as follows:

$$L_{h,\Psi}(\Omega)\Psi = (\Omega\Psi_{\mathring{y}})_{\mathring{x}} - (\Omega\Psi_{\mathring{x}})_{\mathring{y}}, \tag{11}$$

$$L_{h,\theta}(\Psi)\theta_{k,m} = \frac{1}{2h_1}\left(A_{k+\frac{1}{2},m}(\theta_{k+1,m} + \theta_{k,m}) - A_{k-\frac{1}{2},m}(\theta_{k,m} + \theta_{k-1,m})\right) -$$

$$-\frac{1}{2h_2}\left(B_{k,m+\frac{1}{2}}(\theta_{k,m+1} + \theta_{k,m}) - B_{k,m-\frac{1}{2}}(\theta_{k,m} + \theta_{k,m-1})\right),$$

where

$$A_{k+\frac{1}{2},m} = \frac{1}{4h_2}(\Psi_{k+1,m+1} + \Psi_{k,m+1} - \Psi_{k+1,m-1} - \Psi_{k,m-1}),$$

$$B_{k,m+\frac{1}{2}} = \frac{1}{4h_1}(\Psi_{k+1,m+1} + \Psi_{k+1,m} - \Psi_{k-1,m+1} - \Psi_{k-1,m}).$$

The following estimate holds for the solutions of the difference problem (5)-(8):

$$\|\Delta_h\Psi\| \leq \frac{1}{\delta_0}\left(\frac{\mathrm{Ra}}{\sqrt{\delta_0}}\|g\| + \|f\|\right), \tag{12}$$

where $\delta_0 > 0$ is the minimal eigenvalue of the Laplace difference operator, $\mathrm{Ra} = \mathrm{Gr}\,\mathrm{Pr}$ is the Rayleigh number, $\|f\|$ is norm of the grid function in space $L_{2,h}(D_h)$.

Indeed, multiplying the relation (5) by Ψ and summing over the inner nodes of the grid D_h, we obtain the relations:

$$(L_h(\Omega)\Psi,\Psi) = (\Delta_h\Omega,\Psi) + (A_h\Psi,\Psi) + \mathrm{Gr}\,(\theta_{\mathring{x}},\Psi) + (f,\Psi),$$

$$\|\Delta_h \Psi\|^2 + \sum_{m=1}^{N_2-1} \left(\Omega_{0m}\Psi_{x,0m} - \Omega_{N_1 m}\Psi_{\bar{x},N_1 m}\right) h_1 + \sum_{k=1}^{N_1-1} \left(\Omega_{k0}\Psi_{y,k0} - \Omega_{kN_2}\Psi_{\bar{y},kN_2}\right) h_2 +$$

$$+ \mathrm{Gr}\left(\theta_{\mathring{x}}, \Psi\right) + (f, \Psi) = 0.$$

Hence, taking into account the boundary conditions (8), we have:

$$\|\Delta_h \Psi\|^2 + \frac{2}{h_1}\sum_{m=1}^{N_2-1}\left(|\Psi_{x,0m}|^2 + |\Psi_{\bar{x},N_1 m}|^2\right)h_1 + \frac{2}{h_2}\sum_{k=1}^{N_1-1}\left(|\Psi_{y,k0}|^2 + |\Psi_{\bar{y},kN_2}|^2\right)h_2 \leq$$

$$\leq \mathrm{Gr}\left|(\theta_{\mathring{x}}, \Psi)\right| + |(f, \Psi)|,$$

$$\|\Delta_h \Psi\|^2 \leq \mathrm{Gr}\left|(\theta_{\mathring{x}}, \Psi)\right| + |(f, \Psi)|.$$

Further, the right-hand side of the inequality obtained is estimated as follows:

$$\mathrm{Gr}\left|(\theta_{\mathring{x}}, \Psi)\right| \leq \mathrm{Gr}\|\nabla_h\theta\|\,\|\Psi\| \leq \frac{\mathrm{Gr}}{\delta_0}\|\nabla_h\theta\|\cdot\|\Delta_h\Psi\|,$$

$$|(f, \Psi)| \leq \frac{1}{\delta_0}\|\Delta_h\Psi\|\cdot\|f\|,$$

and we obtain the estimate

$$\|\Delta_h\Psi\| \leq \frac{1}{\delta_0}\left(\mathrm{Gr}\|\nabla_h\theta\| + \|f\|\right). \tag{13}$$

Further, multiplying the relation (7) by θ and summing over the inner nodes of the grid D_h, we obtain the following energy identity:

$$\left(L_{h,\theta}\left(\Psi\right)\theta, \theta\right) = \frac{1}{\mathrm{Pr}}\left(\Delta_h\theta, \theta\right) + (g, \theta).$$

Hence we have

$$\frac{1}{\mathrm{Pr}}\|\nabla_h\theta\|^2 \leq |(g, \theta)| \leq \frac{1}{\sqrt{\delta_0}}\|\nabla_h\theta\|\cdot\|g\|,$$

i. e.

$$\|\nabla_h\theta\| \leq \frac{\mathrm{Pr}}{\sqrt{\delta_0}}\|g\|. \tag{14}$$

Using (13) and (14), we obtain the desired estimate (12).

Let us introduce the auxiliary grid function of the velocity vortex as in [9, 10], and represent the equations (5)-(7) in the form:

$$L_{h,\Psi}\left(\Omega\right)\Psi = \Delta_h\Omega + A_h\Psi + \mathrm{Gr}\theta_{\mathring{x}} + f, \tag{15}$$

$$\Delta_h\Psi = \Omega, \tag{16}$$

$$L_{h,\theta}\left(\Psi\right)\theta = \frac{1}{\mathrm{Pr}}\Delta_h\theta + g \tag{17}$$

with homogeneous boundary conditions for the sought grid functions, including the velocity vortex; A_h is a nonnegative difference operator.

In the case of choosing $L_{h,\Psi}$ by (11), the operator A_h has the following form [9]:

$$A_h \Psi_{km} = \begin{cases} \dfrac{2}{h_1^4} \left(\delta^{k,1} + \delta^{k,N_1-1} \right) \Psi_{km}, & k = 1, N_1 - 1, \quad m = \overline{1, N_2 - 1}, \\ \dfrac{2}{h_2^4} \left(\delta^{m,1} + \delta^{m,N_2-1} \right) \Psi_{km}, & k = \overline{1, N_1 - 1}, \quad m = 1, N_2 - 1, \end{cases}$$

where $\delta^{k,m}$ is the Kronecker symbol.

For numerical solution of the difference problem (15)-(17) we consider an iterative algorithm of the variable direction type:

$$\frac{\Omega^{n+\frac{1}{2}} - \Omega^n}{\tau} + L_{h,\Psi} \left(\Omega^n \right) \Psi^{n+\frac{1}{2}} = \Delta_h \Omega^n + A_h \Psi^{n+\frac{1}{2}} + \mathrm{Gr}\theta_{\mathring{x}}^{n+1} + f, \qquad (18)$$

$$\Delta_h \Psi^{n+\frac{1}{2}} = \Omega^{n+\frac{1}{2}}, \qquad (19)$$

$$\frac{\Omega^{n+1} - \Omega^{n+\frac{1}{2}}}{\tau} = \Delta_h \left(\Omega^{n+1} - \Omega^n \right), \qquad (20)$$

$$\Delta_h \Psi^{n+1} = \Omega^{n+1}, \qquad (21)$$

$$L_{h,\theta} \left(\Psi^n \right) \theta^{n+1} = \frac{1}{\mathrm{Pr}} \Delta_h \theta^{n+1} + g, \qquad (22)$$

$$\Psi^0 \left(x, y \right) = \Psi_0 \left(x, y \right) = 0, \quad \Psi^{n+1} = \Omega^{n+1} = \theta^{n+1} = 0, \quad (x, y) \in \partial D_h. \qquad (23)$$

To implement the solutions of the auxiliary grid equations with nonself-adjoint operators (18) and (22), we use the modified method of minimal corrections of the variational type.

To solve the equations (20), (21), we apply the classical alternating-triangular method with the Chebyshev set of parameters [8].

Numerical Results

The following test problems are solved for the computational experiments of the algorithm, the modified method of minimal corrections of the variational type and the iterative algorithm (18)-(23):

- the problem of a cavity;
- the problem of free convection in a closed domain when heated from above.
- the problem of free convection in a closed domain when heated from the side.

Computational calculations were carried out until the following inequalities were satisfied:

$$\left\| L_h \left(\Omega^n \right) \Psi^n - \frac{1}{\mathrm{Re}} \Delta_h \Omega^n - A_h \Psi^n + \frac{\mathrm{Gr}}{\mathrm{Re}} \theta_{\mathring{x}}^n \right\| + \left\| L_h \left(\Psi^n \right) \theta^n - \frac{1}{\mathrm{Re}\,\mathrm{Pr}} \Delta_h \theta^n \right\| \leq \varepsilon,$$

where $\varepsilon = 10^{-4}$.

The Problem of Heating from Above

Testing the problem of heating from above, it can be seen that the number of time iteration remains unchanged with the increase in the size of the grid. The number of iterations increases evenly (almost 0.1 times) with a decrease in the number of steps in time. On a grid with size 33×33, the number of the time iteration is 2648 at $\tau = 0.01$, and the iteration number is 2942 at $\tau = 0.009$ (Table 1). Changing the time step further, a uniform change in the number of iterations can be seen.

Table 1. The problem of heating from above at Re = 400, Gr = 10^3, Pr = 0.73, $\varepsilon = 10^{-4}$

τ	33×33	65×65	129×129
1×10^{-2}	2648	2813	2865
9×10^{-3}	2942	3125	3183
8×10^{-3}	3309	3516	3581
7×10^{-3}	3782	4019	4092

The Problem of a Cavity

When testing cavity problems, the number of time iteration decreases uniformly with the increase in the grid size. The number of iterations increases uniformly on reducing the time step. On a grid with size 33×33, the number of the time iteration is 3905 at $\tau = 0.01$, and the iteration number is 4339 at $\tau = 0.009$ (Table 2). Thus, changing the time step further, a uniform change in the number of iterations can be seen.

Table 2. The problem of a cavity at Re = 400, Pr = 0.73, $\varepsilon = 10^{-4}$

τ	33×33	65×65	129×129
1×10^{-2}	3905	3845	3838
9×10^{-3}	4339	4271	4262
8×10^{-3}	4881	4805	4792
7×10^{-3}	5579	5491	5475

The Problem of Heating from the Side

In this problem, the number of time iteration increases almost threefold as the grid size increases. The number of iterations changes uniformly on changing the

τ	33×33	65×65	129×129
1×10^{-2}	377	923	3076
9×10^{-3}	490	1105	3504
8×10^{-3}	727	1451	4268
7×10^{-3}	1598	2670	6886

Table 3. The problem of heating from the side at $Gr = 10^5$, $Pr = 0.73$, $\varepsilon = 10^{-4}$

time step. On a grid with size 33×33, the number of the time iteration is 377 at $\tau = 0.0004$, and the iteration number is 490 at $\tau = 0.0003$ (Table 3).

Figure 1 shows flow patterns in the form of an isotherm and a streamline of the problem of free convection in a square domain when heated from the side at values $Gr = 10^5$, $Pr = 0.73$, $\varepsilon = 10^{-4}$.
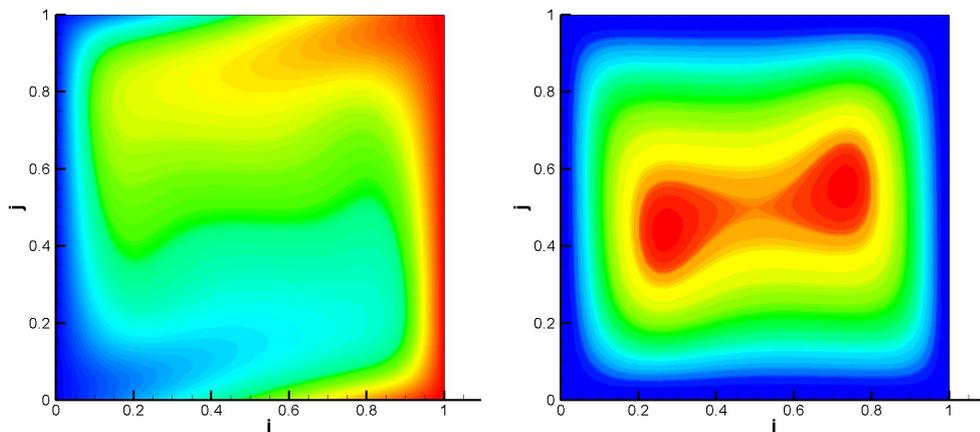
Figure 1. Isotherms and stream lines of the problem of free convection when heated from the side

Figure 2 shows pictures of the isotherm and streamline of the problem of convective flow in a square domain with a moving cover for different Reynolds numbers and $Gr = 2 \times 10^5$.

Conclusion

According to the results of the study, we can conclude that the use of the algorithm (18)-(23) for solving equations with nonselfadjoint operators is effective. It can be concluded from the analysis of the computational experiment that the proposed and mathematically based iterative algorithms are viable and can be used to solve the Navier-Stokes equations numerically.
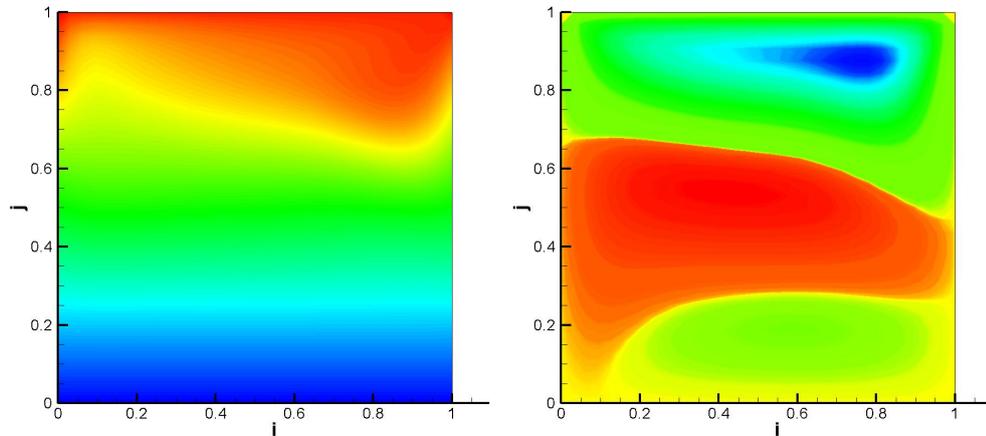
Figure 2. Isotherms and streamlines at Re = 500

References

1. Voevodin, A.F.: Ustojchivost i realizaciya neyavnykh skhem dlya uravnenii Stoksa. Zhurnal vychislitelnoi matematiki i matematicheskoi fiziki 33, No. 1, 119–130 (1993)
2. Voevodin, A.F., Yushkova, T.V.: Chislennyi metod resheniya nachalno-kraevykh zadach dlya uravnenii Navie-Stoksa v zamknutykh oblastyakh na osnove metoda rasshhepleniya. Sibirskii zhurnal vychislitelnoi matematiki 2, No. 4, 321–332 (1999)
3. Danaev, N.T., Amenova, F.S.: Issledovanie skhodimosti iteracionnykh algoritmov chislennogo resheniya zadach teplovoi konvekcii v peremennykh «funkciya toka, vixr skorosti». SibZhIM 17, No 3 (59), 48–58 (2014)
4. Amenova F.S.: Chislennoe reshenie setochnykh uravnenii Navie-Stoksa dlya neszhimaemoi zhidkosti v peremennykh «zavixrennost, vektornyi potencial», PhD thesis, al-Farabi, Almaty (2015)
5. Tarunin, E.L.: Vychislitelnyi eksperiment v zadachakh svobodnoi konvekcii. Irkutsk, Izd. Irkut. un-ta (1990)
6. Roach, P.: Vychislitelnaya gidrodinamika. Moscow, Mir (1980)
7. Temam, R.: Uravneniya Navie-Stoksa: teoriya i chislennyi analiz. Moscow, Mir (1981)
8. Samarskii, A.A.: Teoriya raznostnykh skhem. Moscow, Nauka (1983)
9. Vabishhevich, P.N.: Realizaciya kraevykh uslovii pri reshenii uravnenii Navie-Stoksa v peremennykh (ψ, ω). Dokl. AN SSSR 273, No 1, 22–26 (1983)
10. Danaev, N.T., Smagulov, Sh.: Ob odnoi metodike chislennogo resheniya Navie-Stoksa v peremennykh (ψ, ω). Modelirovanie v mekhanike 5 (22), No. 4, 38–47 (1991)

Construction of unstructured grids on oil and gas fields with complex form

N. Danayev , O. Turar, and D. Akhmed-Zaki

Al-Farabi Kazakh National University, Almaty, Kazakhstan
`turar_olzhas@mail.ru,darhan_a@mail.ru`

Abstract The paper describes a method for constructing unstructured grids with adaptation based on differential methods. The application of these methods ensures a smooth distribution of the geometric characteristics of the grid cells. To ensure proper adaptation in unstructured grids, we use a new approach based on the methodology for constructing an adaptive structured grid. This approach uses the method of constructing a grid based on the solution of the inverted Beltrami equation to create a mapping of some area of the sample grid to the physical domain.

Keywords: computational grid construction algorithm, unstructured mesh, adaptive mesh, differential elliptic equations, reversed Beltrami equation.

Introduction

Normally, the grids constructed for the numerical solution of partial differential equations are distributed into two essential groups: structured and unstructured grids [1]. Each of these groups has its own methodology of construction and use, which almost does not overlap. However, some theoretical principles that can be realized for general numerical grids allow us to consider the possibility of comparing and combining some methods from different groups. One of them is the adaptation of grids.

Adaptive grids allow researchers to avoid creating fine grids for the entire domain. Such grids critically increase the execution time of calculations. In many cases, locations with a smooth solution do not need this excessive accuracy, because it does not take effect in view of the poor accuracy of places with high gradients of the solution.

Computational grids can be constructed in such a way that the places of such large solution values and gradients are covered by small cells. This increases the accuracy of the numerical solution created in places of steep gradients and high values. The grid in the places of smooth solutions coarsens to optimize the computation time.

Unstructured grids are widely used in finite element and finite volume methods. They are necessary in certain situations and have several advantages

over structured grids in the case of complex domain forms. Basically, such grids are constructed by geometric and graph methods. Adaptation in these methods is carried out by adding additional points in certain places of the grid. Most algorithms for generating an unstructured grid [2] belongs to one of 3 groups, defined by base approach to the construction. The first group is advancing front algorithms [3]. In such algorithms, the mesh constructed by attaching new cells to an existing grid, starting from the boundaries of the region. The next approach is the iterative construction of the Delaunay triangulation by adding new points to the existing triangulation [4, 5]. After adding a new point, the entire triangulation must be changed to satisfy the Delaunay's principle. Algorithms based on trees, such as the quadtree [6, 7] algorithms. In such algorithms, the lattice cell shapes of the results may be far from perfect. However, this approach makes it very convenient to adapt the grid in terms of cellular blackening. These approaches can be improved to support limited grids, such as constrained Delaunay triangulation [8, 9]. Limited grids are meshes that contain some initial predefined structural elements, such as pre-initialized curves or edges, surfaces, and volumes. This is how we are going to add flaws and fractures in the grid.

General structured grids, commonly used in methods with finite difference, are constructed using simple algebraic methods. But their design is complicated in situations of adaptation to the boundaries of physical regions and control functions. In this case, the structured grid is a curvilinear coordinate system and is determined by solving nonlinear differential equations. Such methods are based on numerical differential geometry and lead to very natural adaptation of structured grids [15, 16, 17]. There are lot of different approaches of constructing structured adaptive grids [18, 19, 20, 21].

The article describes a method of constructing a grid that has positive characteristics of both approaches, combining them into one method. In the case of unstructured grids at the moment there are several approaches [10, 11, 12, 13, 14. But they are all based on algebraic methods. In the general case, it is difficult to guarantee algebraic methods such good grid characteristics as smoothness and regular cellular forms. The best numerical grids have to retain the similarity of the shapes and sizes of adjacent cells throughout the domain. For this purpose, methods based on differential equations can be very useful. The combination is carried out by constructing an unstructured grid based on a set of points uniformly distributed over a curvilinear structured grid. In this method, the resulting grid is an unstructured grid and, therefore, has all its positive aspects. It also maintains a smooth adaptation based on differential methods.

Namely, the approach we are going to use is solving the inverted Beltrami equation [18, 16]. This approach guarantee the property of smoothness for cell form and size changes since it is based on variational methods. The equation is taken by minimization of energy functional which is based on theory of constructing isometric coordinate system on the surface [15]. Thus, the approach is the best in terms of the physicality of the result grids.

Grid construction method

Adaptive structured grids

Structured grid adaptation are based mainly on solving differential elliptic equations. Consequently, they tend to smooth the scattering of the grid nodes. They find a mapping of a sample domain with a common Cartesian grid into a given physical region (Fig. 1, A). One of the most advanced in terms of adaptation is the method based on the solution of the inverted Beltrami equation (not inverse), which can be used to construct structured grids adapted to gradients or values of scalar fields or to directions of the vector field [18, 16]:

$$\frac{\partial}{\partial s^j}(\sqrt{g^s}g_s^{jl}) = \sqrt{g^s}g_s^{im}\frac{\partial^2 s^l}{\partial \xi^i \partial \xi^m} \tag{1}$$

In the formula (1), repeating indexation means summation over those indexes on the one side of equality. S represents new grid coordinate system and ξ^i is initial grid's curvilinear coordinate system. Here $g_s^j l$ are contravariant tensor components and g^s is its determinant of the mapping shown in figure 1, A. Equation (1) is non-linear in such form and cannot be solved using standard elliptic equation solving methods, so to solve it we have to add time derivative. Thus, we convert it into parabolic equation and solve it by iterations. In each iteration we get new curvilinear grid which is little closer to the result adaptation than the previous one. Boundary conditions in those methods are taken by solution of the same problem on the lesser dimension (figure 1, B and C).

Unstructured grids

To construct unstructured grids, we used the Delaunay triangulation and the Voronoi diagram [5], since they create relatively good grids for computation by a given set of points oriented to control volume methods. The Delaunay triangulation is a triangulation of the set of points in which all triangles and points satisfy the Delaunay principle (Fig. 2, A). This principle states that triangles should not contain other points from the set inside their circumcircles. The goal is to create such grids in arbitrary cases with initial structural constraints, such as predefined edges that can not be reconstructed.

Diagram of Voronoy is defined explicitly from triangulation by changing points to cell centres and cell centres to nodes. Such grids are very useful to control volume methods when centres of Voronoy cells can be counted as their geometrical centres. In such cases the segments connecting centres of adjacent cells are orthogonal to the edge or face between them.

To choose a set of points on which we implement the unstructured grid construction method described above is to use the curvilinear adaptation provided by differential methods. The set is uniformly scattered on the sample region (left part of Fig. 1, A). After implementation of the mapping, we gain uniformly scattered set with the adaptation to the values of some scalar control
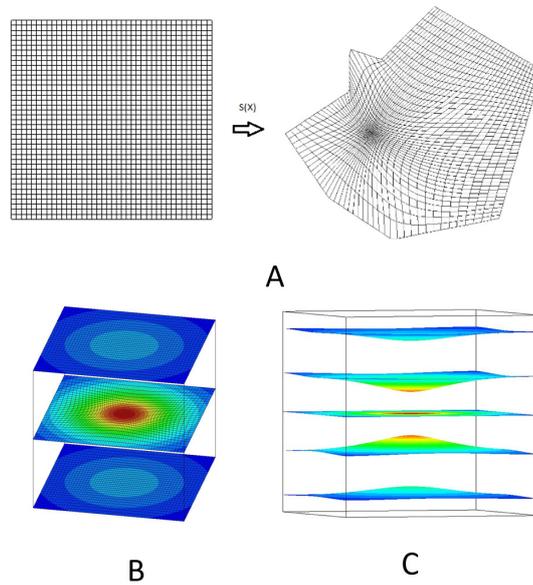
Figure 1. Structured grid construction: A) the mapping to be found; B-C) reverted Beltrami equation result.

function. Further, this adapted set of points is used to construct an unstructured grid. Figures 3 and 4 show the results of the implementation of the proposed algorithm. The points in Figure 3 are selected simply as nodes of the structured grid after adaptation. Since the sample cells are squares, the adaptive grid lines are still close to orthogonal. Therefore, in the triangulation shown in Figure 3, B, we see that the triangles are close to the right triangle.

Our way to choose the point set is to use curvilinear adaptation provided by differential methods. The set is uniformly scattered on the sample domain, which is represented on the left side of figure 2, A. So, after mapping it represents uniformly scattered set with adaptation to the values of some scalar control function. Further this adapted set of points is used to construct the unstructured grid. Figures 3 demonstrates the results of the implementation of proposed algorithm.

Discussion

In real industrial physical modeling applications, the process domain can have very complex shapes. They can have internal constraint structures, such as faults or fractures, or a very complex boundary topology. In such cases, it is very difficult to use structured meshes even with adaptation. Complex boundaries
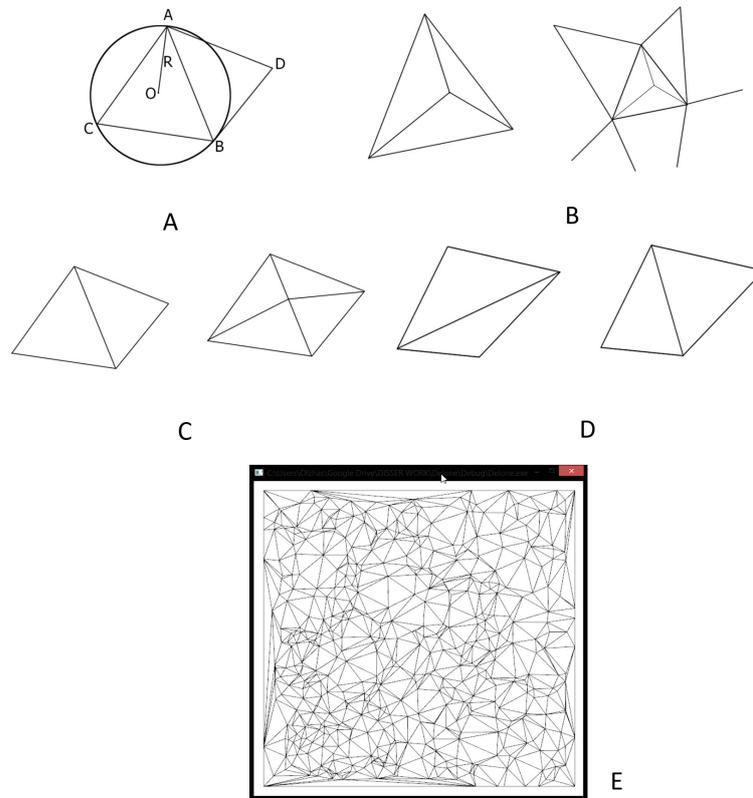
Figure 2. Delaunay triangulation: A) Delaunay principle; B) Delaunay construction: adding new point in triangular cell; C) Delaunay construction: adding new point to existing edge; D) flip operation; E) demonstration of constructed triangulation on the random 100 points.

force researchers to make extremely small grids near these boundaries in structured grids. In the case where there are structural constraints that establish limitations for building a computational grid unstructured grids have special tools like constrained Delaunay triangulations. These structural lines or surfaces can intersect in a random order and create a complex unstructured topology making impossible their accurate tracing by structured grid lines.

These advantages of unstructured grids are preserved in our method, since result grids of the proposed method is an unstructured grid. The main advantage of structured grids is the adaptation behaviour provided by differential elliptic methods. This behaviour preserved in our method. This leads to smooth changes in such properties of the grid cell as the size and shape, i.e., how much the cell is elongated in one dimension.
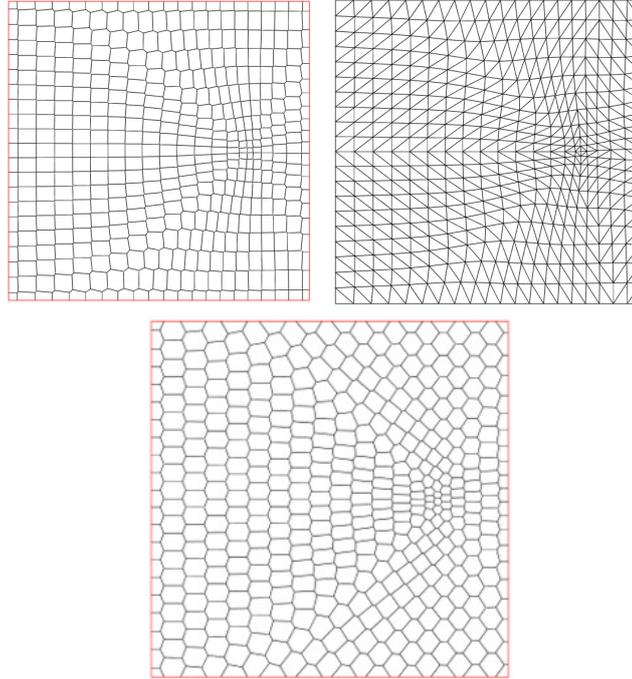
Figure 3. Example of unstructured grid constructed using points set defined by adaptive mapping: upper left and bottom – Voronoy diagrams and upper right – Delaunay triangulation.

In contrast to the existing methods of adapting an unstructured grid, that are based on the use of geometry and graph algorithms, differential methods based on elliptic equations guarantee such smoothness. It is very unnatural and inconvenient fulfil the necessity of smoothness by algebraic methods. But the elliptic differential methods has fundamental property to even the values of the solution. In addition, the method of solving the inverse Beltrami equation is based on minimizing the energy functional.

One significant drawback of curvilinear structured grids is its use with finite difference methods. Since the computational grid on physical domain is a mapping of some sample grid researchers must bring the expression to the correct form. Expressions are to be modified by inserting of the mapping to the equation which lead to Jacobians of the mapping and components of the metric tensor in the final form of the mathematical model. Simple heat transfer equation after such modification is written in the following form

$$\frac{\partial u}{\partial t} = \nu \left(\frac{\partial}{\partial \xi^1} (\sqrt{g} g^{11} \frac{\partial u}{\partial \xi^1} + \sqrt{g} g^{12} \frac{\partial u}{\partial \xi^2} + \sqrt{g} g^{13} \frac{\partial u}{\partial \xi^3}) + \right.$$

$$\frac{\partial}{\partial \xi^2}(\sqrt{g}g^{21}\frac{\partial u}{\partial \xi^1} + \sqrt{g}g^{22}\frac{\partial u}{\partial \xi^2} + \sqrt{g}g^{23}\frac{\partial u}{\partial \xi^3}) + \qquad\qquad (2)$$

$$\frac{\partial}{\partial \xi^3}(\sqrt{g}g^{31}\frac{\partial u}{\partial \xi^1} + \sqrt{g}g^{32}\frac{\partial u}{\partial \xi^2} + \sqrt{g}g^{33}\frac{\partial u}{\partial \xi^3}))$$

Solving even sample equations may bring challenge in case of adaptive structured grids. Since all that process is not yet automatized and have to be done manually using unstructured grids on the same data

Conclusion

The paper describes novel approach method that keeps smoothness property of adaptive structured grid and transfers it to unstructured grids. By using smooth mapping produced by structured grid construction method we initialize set of points on which the unstructured grid constructed. The grids are Delaunay triangulations and Voronoy diagrams and constructed iterationally. The point set for construction acquired by mapping of some regular triangular nodes using smooth mapping, that have been defined by solving elliptical differential equations. Due to that fact inner angles of cells are close to each other in the whole triangulation. Exceptions may occur if we need to add structural constraints that are not traced by grid lines of coordinate system that represents mapping, because in such case initial point set has not any kind of relation to the predefined structural line or surface.

However, the most important reason to use adaptive structured grids is a possibility to use finite differences method with higher order approximation difference schemes. Proposed approach does not interact with such situations and may be handy only in situations when those schemes are not necessary.

Differential equations well fit situations when changes of cells size and form is to be smooth. Until now, methods of unstructured adaptive grid construction have used mainly algebraic formulas which leads to necessity of some additional instrument to guarantee of such smoothness. In case of differential methods based on elliptic equations such property is encapsulated in its basic principles. In grid construction methodology such methods have been used widely to construct structured adaptive grids. To transfer this property to unstructured meshes, we implemented standard grid construction algorithms with the ability to build a grid on any given set of points. Solving the equation provides a certain mapping of some sample grid to the physical domain, and we use this mapping to define a set of points with the adaptation. On this set the unstructured grid is constructed which gives us the result grids that have benefits of both approaches.

References

1. Thompson, Joe F., Soni, B. K. and Weatherill, N. P.: Handbook of Grid Generation. Boca Raton, CRC Press, FL, (1999).

2. Shewchuk, J. R.: Unstructured mesh generation. In U. Naumann and O. Schenk, editors, Combinatorial Scientific Computing. , chapter 10, 257–297, CRC Press (2012).
3. Lo, S. H.: A New Mesh Generation Scheme for Arbitrary Planar Domains. International Journal for Numerical Methods in Engineering. 21, no. 8 (1985). doi:10.1002/nme.1620210805.
4. Shewchuk, J.R.: Tetrahedral mesh generation by Delaunay refinement. Symp. on Comp. Geom. , 86–95, (1998).
5. Klein, Rolf, Aurenhammer, Franz and Lee, Der-Tsai: Voronoi Diagrams and Delaunay Triangulations. , World Scientific, New Jersey (2013).
6. Yerry, Mark, and Shephard, Mark: A Modified Quadtree Approach To Finite Element Mesh Generation. IEEE Computer Graphics and Applications. 3, no. 1, 39-46 (1983). doi:10.1109/mcg.1983.262997.
7. Mitchell, Scott A., and Vavasis, Stephen A.: Quality Mesh Generation in Three Dimensions. Proceedings of the Eighth Annual Symposium on Computational Geometry - SCG. 92 (1992). doi:10.1145/142675.142720.
8. Frederick, C. O., Wong, Y. C. and Edge, F. W.: Two-dimensional Automatic Mesh Generation for Structural Analysis. International Journal for Numerical Methods in Engineering 2, no. 1 (1970). doi:10.1002/nme.1620020112.
9. Lee, Der-Tsai and Lin, Arthur K.: Generalized Delaunay Triangulations for Planar Graphs.' Discrete & Computational Geometry, 1, 201–217 (1986).
10. Bronson, Jonathan R., Sastry, Shankar P., Levine, Joshua A. and Whitaker, Ross T.: Adaptive and Unstructured Mesh Cleaving. Procedia Engineering. 82 (2014). doi:10.1016/j.proeng.2014.10.389.
11. Riemslagh, K., and Dick. E.: A multigrid method with unstructured adaptive grids for steady Euler equations. Journal of Computational and Applied Mathematics 67, 73-93 (1996).
12. Vilsmeier, R., et all: Adaptive Methods on Unstructured Grids for Euler and Navier-Stokes Equations. Computers & Fluids. 22, no. 4-5 (1993). doi:10.1016/0045-7930(93)90021-z.
13. Bronson, Jonathan R., Levine, Joshua A. and Whitaker, Ross T.: Particle Systems for Adaptive, Isotropic Meshing of CAD Models. Proceedings of the 19th International Meshing Roundtable (2010)
14. Mavriplis, D. J.: Unstructured Grid Techniques. Annual Review of Fluid Mechanics. 29, no. 1, 473-514 (1997). doi:10.1146/annurev.fluid.29.1.473.
15. Vekua, I. N.: Generalized Analytic Functions. Pergamon Press, Oxford (1962).
16. Lisejkin, V. D.: Grid Generation Methods. Springer, Cham (2017).
17. Castillo, Jose E.: Mathematical Aspects of Numerical Grid Generation. SIAM, Philadelphia (1991).
18. Glasser, A. H., Liseikin, V. D. and Kitaeva, I. A.: Specification of Monitor Metrics for Generating Vector Field-aligned Numerical Grids. Russian Journal of Numerical Analysis and Mathematical Modelling. 20, no. 5, 439-461 (2005). doi:10.1515/156939805775186686..
19. Godunov, S. K., Zabrodin, A. V. and Ivanov, M. Ja.: Chislennoe reshenie mnogomernyh zadach gazovoi dinamiki (Numerical solving of multidimensional problems in gas dynamics). Nauka, Moscow (1976).
20. Khakimzyanov, G. S., and Shokina, N. Yu.: Equidistribution Method for the Construction of Adaptive Grids. Russian Journal of Numerical Analysis and Mathematical Modelling 14, no. 4, 339-358 (1999). doi:10.1515/rnam.1999.14.4.339.
21. Thompson, Joe F., Warsi, Z. U. A. and Wayne, C.: Mastin. Numerical Grid Generation: Foundations and Applications. North-Holland, New York (1985).

Mathematical model development for visual quality control of coatings sprayed on the products for medical purposes

M. Iskakova, E. Ryzhkova, and O. Baklanova

D.Serikbayev East Kazakhstan State Technical University, School of Information Technology and Energy,
Serikbayev street 19, 070010 Ust-Kamenogorsk, Kazakhstan
`madina.iskakova13mk@gmail.com`

Abstract Nowadays, there is significant progress in areas of medicine, connected with using of implants (surgery, orthopedics, endoprothesis replacement) due to development of making implants technology and medical tools which is necessary to their installation. One of the main conditions of successful functioning such products for medical purposes is ensuring their bio-compatibility. In this article we will consider the most common approaches to the texture descriptions and on the basis of the obtained theoretical data we will choose the method which will allow estimating quality of texture in any scale of the image.

Keywords: texture, bio-compatible coating, Gabor filter

Introduction

Nowadays, we can observe significant progress in the areas of medicine, concerning implants usage (surgery, orthopedics, endoprosthesis replacement) due to development of making implants technology and medical tools, necessary for their adjusting. One of the main factors of successful functioning of such products for medical purposes is ensuring their bio-compatibility. For reaching this goal, we should satisfy two conditions:

1. Proper selection of implant's material;
2. Different technologies of implant's bio-compatibility increasing by means of various types of their surface treatment including bio-compatible coatings spraying [1, 2].

There are several high-tech devices, which are used as biocompatible coatings deposition on implants, but also they cannot guarantee an ideal covering. To guarantee that the implant would not be rejected, we will consider various methods of the texture descriptions for the subsequent analysis of the image.

R.M. Kharalik's [3], K.I. Lavs's [4], X. Tamura's and [5], X. Niemann's [6] studies are devoted to automation of processing, analysis, recognition and comprehension of textural features. It is essential to mention the works of D.

Chetverikov [7] (a number of algorithms for the selection and classification of textures were investigated), N.G. Fedotov [8] (application of stochastic geometry and functional analysis for the recognition of half-tone textures), K.N. Spiridonov [9] (application of the spectrum of generalized fractal attributes for texture comparison), A.A. Potapov [10] (technology of radar detection of low-contrast targets based on probabilistic textural attributes), and M. Stesheletsky [11] (application of texture analysis to the processing of biomedical images). [12] In this paper, we will consider the most common approaches to the description of textures and basing on the theoretical data, we will choose a method assessing the quality of the texture at any scale of the image.

The Main Approaches to the Description of Textures for Visual Quality Control of Bio-compatible Coatings on Medical Products Made from Titan

Statistical method: this approach allows describing image's texture effectively because of determination of two-dimensional contiguity of pixels' brightness. Advantage of this approach is that textural signs are calculated by the fragment of image. The disadvantage is that this approach is difficult in calculation [3, 13].

Geometrical approach: it is convenient to estimate the repeating character of an arrangement of textural elements of the image by means of function of autocorrelation, which can be used to assess the degree of a regularity and fineness of rough texture morphology. For coarse-grained texture, the size of auto correlated function decreases slightly in comparison with area of fine-grained texture. The advantages of this approach is that methods of stochastic geometry could decrease the bug of autocorrelation function and that Blur-invariants relating to the theory of invariant moments are tolerant to violations of focus of the camera and vibrations of sensors. Disadvantage is that this algorithm is very limited in practical use [14].

Spectral approach: for processing images in different scales Fourier and wavelet analysis are implied. The Fourier transformation should be considered as a decomposition of the signal into frequency and amplitude components, i.e. reversible transformation from temporal to frequency space. In case of wavelet analysis, the signal is decomposed by basic functions (wavelets), which are based on the generating wavelet using shift and scaling operations. The advantage of this approach is that independent component analysis filters reflect the main directions of the image texture and when analyzing medical images, there are noticeable textural deviations that indicate a deviation from the norm. Disadvantage is the difficulty when changing the period and phase of the texture [15].

As the outcome of the studies being carried out, we have established the structural-phase composition and some properties of samples of bars from titanium alloy Grade 5 Eli brand, and Ti6 -Al-4V transmission electron microscopy (TEM) and x-ray analysis revealed that the samples of alloy Grade

5 Eli brands have a two-phase $\alpha + \beta$ microstructure consisting of Ti- α phase with a hexagonal crystal lattice and a small fraction of the Ti β -phase with a cubic crystal lattice.

Figure 1 shows the metallography of alloy samples with ideal and damaged texture.

<center>a)   b)</center>
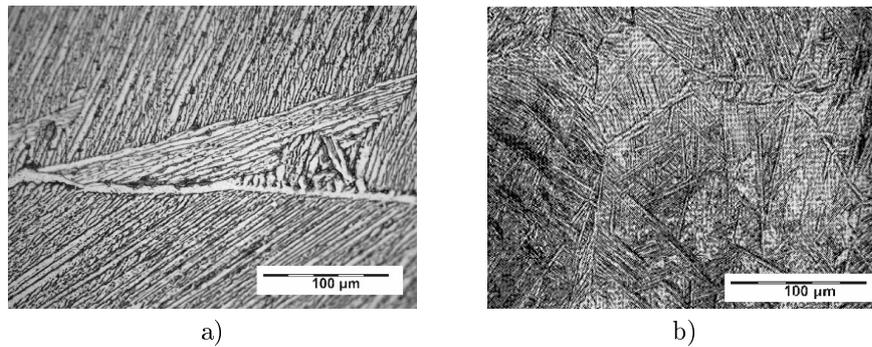
Figure 1. Microstructure of the studied samples of Grade 5 alloy Eli with ideal texture (*a*) and damaged texture (*b*).

Development and growth of α-phase secretions from β, which is known to occur in these alloys during cooling from the temperatures of hot thermomechanical treatment, are presented in the form of strips or plates. The lamellar microstructure consists of parallel lamellae of the α-phase separated by a phase β. The width of the plates varies from two to three dozen to several microns. The grain sizes from 100 to 500 microns. Packages or colonies of α-plates have dimensions from 10 to 500 microns (see figure 1 a) and are located mainly at indirect angles to grain boundaries, but in figure 1 (b) there is a texture damage.

Spectral Approach. Gabor Filter. Algorithm of Gabor Filter

Fourier and wavelet analysis are used to process images in different scales. The Fourier transformation should be considered as a decomposition of the signal into frequency and amplitude components, i.e. reversible transformation from temporal to frequency space. In the case of images, the Fourier transform is also used to identify periodic structures that can be used to detect objects.

The input data is a two-dimensional image with superimposed noise. Suppose you have the input image I size $N \times M$, where N and M - its height and width, then the two-dimensional representation of the signal in the frequency domain occurs according to the formula 1 [16]:

$$\overline{I}[i_1, i_2] = \frac{1}{N * M} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{M-1} I[k_1, k_2] * exp(j * i_1 * k_1 * \frac{2 * \pi}{N} + j * i_2 * k_2 * \frac{2 * \pi}{M}), j = \sqrt{-1}$$

(1)

In case of wavelet analysis, the signal is decomposed by basis functions (wavelets), which are based on the generating wavelet using shift and scaling operations. For the initial image, a convolution with a basic function (filter) is constructed, then the difference between the received and the original signal is calculated and a second projection of this difference is constructed, each basic function is a shift of the previous one stretched by a certain number of times.

Independent component analysis filter and Gabor filter are recognized as the most effective filters [17]. The first ones are obtained by analyzing the training sample images, these filters are local and similar to the Gabor filter, and however, they reflect the main directions of the image texture and are of natural character.

Gabor Filter

The algorithm based on Gabor's filters is very useful in identifying the direction of the texture. This is the main advantage of the algorithm. In digital image processing, this filter is applied to detect edges of objects. RGB and HSL color models are suitable to use this algorithm. It should be noted that this algorithm is not as popular as, for example, graph theory algorithms. Therefore, the algorithm based on Gabor filters is of the greatest interest for its development and further implementation.

The algorithm using Gabor filters is based on the work with two matrices: image matrix and filter matrix. The original image can be represented as a matrix I, the image matrix filled with pixel values. These values are placed in each cell of the matrix. The filter matrix and image matrix are superimposed on each other [18].

The fundamental formula used in the implementation of the algorithm is:

$$g(x, y, \delta, \theta, \phi, \sigma, \gamma) = exp(-\frac{x'^2 + y^2 * y'^2}{2 * \sigma^2}) * cos(2 * \pi * \frac{x'}{\delta} + \phi)$$

(2)

$$x' = x * cos\theta + y * sin\theta$$

(3)

$$y' = -x * sin\theta + y * cos\theta$$

(4)

where,

δ - wavelength of the cosine factor;

θ - determines the orientation of the normal of the parallel bands of the Gabor function in degrees;

ϕ - determines the phase shift in degrees;

γ - compression ratio.

The Algorithm of the Software Product that Implements the Gabor Filter

The algorithm of the software product that implements the Gabor filter is presented in figure 2.
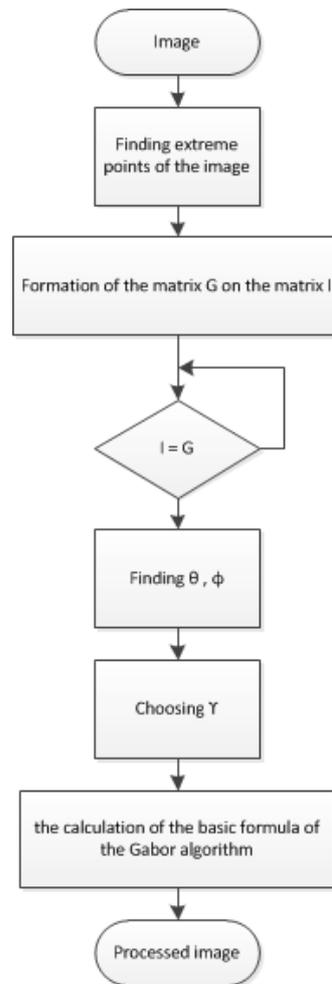
Figure 2. The algorithm of the software product that implements the Gabor algorithm.

Outcome

The initial image samples of the Grade 5 Eli alloy is presented in figure 3. The result of the software is shown in figure 4.

The processed image was obtained with the following data: $\sigma = 8$; $\theta = 0$; $\phi = 1$; $\gamma = 0.5$; N = 8.
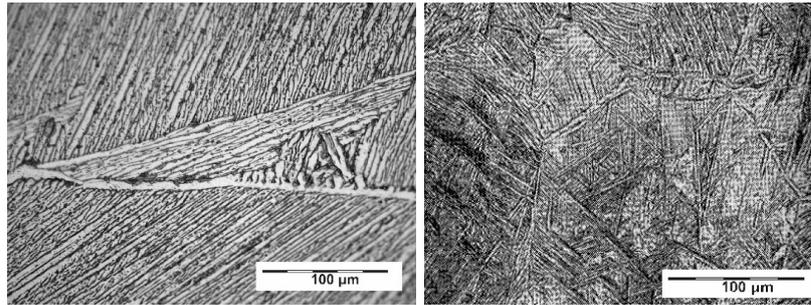
Figure 3. The initial image samples of the Grade 5 Eli alloy.
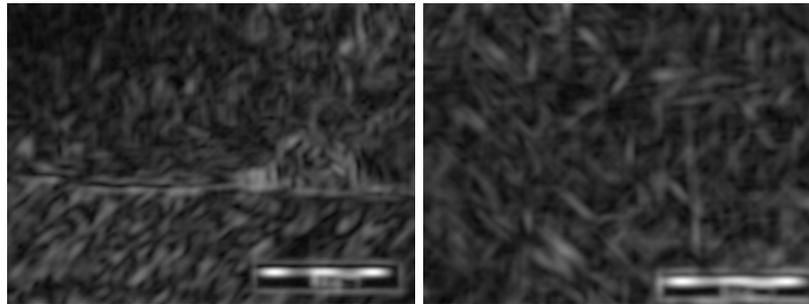
Figure 4. Software result.

Conclusion

In this article, we have considered such approaches to the description of textures as statistical, geometric and spectral. The advantages and disadvantages of each were considered, and basing on the work performed, spectral approach was chosen, which allows assessing the quality of the texture at any scale. Software based on Matlab R2015b was developed as the result of the research. The software allows carrying out quality control of bio-compatible coatings.

Acknowledgment

References

1. Ahsan M.N.: A comparative study of laser direct metal deposition characteristics using gas and plasma-atomized Ti-6Al-4V powders. Materials Science and Engineering, Vol. 528, pp. 7648-7657. (2011)
2. Hohmann M., Pleier S. Production methods and applications for high-quality metal powders and spray formed products. Acta metallurgica sinica, Vol. 18, pp. 15-23. (2005)
3. Haralick R.M. Statistical and structural approaches to texture. Proceedings of the IEEE. Vol.67, no.5, p.768–804. (1979)
4. Laws K.I. Rapid texture identification. SPIE. Vol.238, p.376–380. (1980)
5. Tamura H., Mori S., Yamawaki T. Textural features corresponding to visual perception. IEEE Transaction on Systems, Man and Cybernetics. Vol.8, p.400–473. (1978)
6. Niemann H. Pattern analysis. Springer Series in Information Sciences, Vol.4. Berlin: Springer-Verlag - 302 p. (1981)
7. Chetverikov D. Texture feature based interaction maps and structural filtering // 20th Workshop of the Austrian Pattern Recognition Group, p.143–157. (1996)
8. Fedotov N.G. methods of stochastic geometry in object recognition. Moscow.: Radio and communications, 144 p. (1990)
9. Spiridonov K.N. Application of the spectrum of generalized fractal dimensions of Renyi for comparison of image textures. Petrozavodsk. (2008)
10. Potapov A.A. New information technologies based on probabilistic textural and fractal characteristics in radar detection of low contrast targets. Radio engineering and electronics. Vol. 48. pp. 1101-1119. (2003)
11. Strzelecki M. Segmentation of textured biomedical images using neural networks. PhD Thesis, Technical University of Lodz, Poland, (1995).
12. Fralenko V.P. Methods of the textural analysis of images, data processing of remote sensing of Earth // Vol. 5:4 p. 16-39. (2014)
13. Tamura H., Mori S., Yamawaki T. Textural features corresponding to visual perception. IEEE Transaction on Systems, Man and Cybernetics. Vol.8, p.400–473. (1978)
14. Antoschuk S.G., Serbina N.A. System of recognition of textural images at environmental monitoring. Artificial intelligence. Vol. 4, p.406–413. (2002)
15. Shitova O.V., Pukhlyak A.N., Drob E.M. The analysis of methods of segmentation of textural areas of images in systems of processing of images. Vol. 8 pp. 182-188. (2014)

16. Aphonsky A.A., Dyakonov V.P. Digital analyzers of a range, signal logic. M.: SOLON-press, –– p. 248. (2009)

17. Bell A.J., Senjnowsky T.J., The "independent components" of natural scenesare edge filters. Vision Research. Vol.37, no.23, p.3327–3338. (1997)

18. Kozlova I.V. Texture segmentation of images. Algorithm based on Gabor filters. Vol. 10 pp. 442-444. (2012)

Hardware and software architecture in E-Healthcare

A. Ismukhamedova, I. Uvaliyeva, and S. Belginova

D. Serikbayev East Kazakhstan State Technical University,
Ust-Kamenogorsk, Kazakhstan
saule.belginova@mail.ru

Abstract Healthcare in Kazakhstan is characterized by a full-scale introduction of information systems into the activities of meicine, medical organizations covering all aspects of medicine. At the same time, there is a serious fragmentation and functional incompatibility of existing information systems, such as the information system of compulsory medical insurance, the information system for the collection and analysis of medical statistics, an integrated electronic medical record, information systems for managing health resources. Within the framework of this article, the results of the analysis of the existing hardware and software architecture in E-Healthcare of Kazakhstan are described.

Keywords: E-health, architecture, medical system, e-health concept.

Introduction

In 2013, the State Program "Information Kazakhstan 2020"was adopted, which introduced the term "e-health"(e-health) and identified ways for the further development of health informatization. In accordance with this program, it is expected that the introduction of Information Communication Technologies in the healthcare system will allow to bring the quality of medical care to a new level. E-health technologies will allow monitoring of the population at a distance, better dissemination of information among patients, improving access to health care, especially in remote areas, for people with disabilities and the elderly [1-3].

By 2020, the implementation of e-health of the RK should provide an opportunity to automatically obtain timely, relevant, reliable, and sufficient information to ensure a safe, fair, quality and sustainable health care system that is tailored to the needs of the patient. This will be possible through the fact that all medical organizations and units of the Ministry of Health will have high-speed and secure access to fully interoperable e-Health systems based on paperless technology.

Based on the analysis of the experience of developed countries in the field of creating an e-health system, the strategy for creating a national concept for the transition to the eHealth system can be presented in three steps:

1. Develop a national vision of the health system that addresses the current health issues in the chosen country. An explanation of why such a strategy is needed, what goals are, how the plan will be implemented;

2. Creation of a roadmap for development in this area for the medium term, which, nevertheless, creates an infrastructure for a long time;

3. Create a plan for monitoring the effectiveness of the project, which would take into account possible risks. Attracting investments for a long time [4, 5].

Separately, telemedicine systems, robotic systems, systems for supporting decision-making, electronic libraries are being introduced. In the very near future a breakthrough is expected in the field of implementation of distance education systems for doctors, relevant in the light of the transition to a system of continuing education and accreditation [6].

Conceptual functional architecture

The vision of the general conceptual scheme of functionality ensuring the achievement of e-health objectives is presented in Figure 2. Not all of these elements of the conceptual framework will be directly implemented centrally, which is part of the new implementation strategy aimed at sharing e-health development efforts among a wide range of stakeholders. At the same time, the task of the Ministry of Health of the Republic of Kazakhstan is to provide the conditions for the achievement of this conceptual vision, through a number of mechanisms, including standardization, regulatory legal frameworks, motivation of medical organizations,

In this scheme, there is an integration bus and a single repository that provides centralized storage of medical and non-medical health data, including electronic health passports of every citizen of the Republic of Kazakhstan. The integration bus is an instrument of interaction between various information systems. A single data repository (EPD) is a source of information for the formation of registers for selected priority diseases. In addition, the ERD provides aggregation and storage of information used to make administrative and financial decisions, as well as in government health statistics. Thus, in order to determine the data set to be transmitted to the EPD, standardization of the structure and content of the FTE is required, as well as the analysis and standardization of data necessary for maintaining state statistics and making managerial and financial decisions [4].

The FTE system is a centralized system for storing, processing and transferring FTE of all patients in the country, which will store basic information about patients throughout their life.

The resource management system is an information system that provides registration of providers of medical services and collection of data on technical and economic indicators and material resources of organizations. Management of directories and classifiers is an information system that ensures the unity of identifiers and classifiers for all participants in e-health of the Republic of Kazakhstan through services.

IP Polyclinic is an information system that supports the main processes related to the provision of medical care at the PHC level. Functionally, it is divided into two main groups maintaining an electronic health passport and

electronic medical records of narrow specialists. IP The hospital is an information system that supports the main business processes related to the provision of medical care at the hospital level.

IS Ambulance is an information system that manages calls and brigades, records the causes and results of emergency and emergency care, medical services provided, medications used, access to FTE, records the receipt and expenditure of medicines and medical products.

The IC of the Hospitalization Bureau is an information system that provides the patient with a free choice of a medical organization and the availability of medical services provided during planned hospitalization.

IP Medicines are an information system that provides management of the need, procurement and turnover of medicines and medical devices in the framework of the MSDS at the national and regional levels.

A single payment system is an information system that provides the function of cost recovery and financial decision-making in health care.

Quality management of medical services is an information system that automates the processes of assessing the quality of medical care, assessing the quality of medical care on the basis of the indicator system and ranking of providers of medical services, as well as licensing, accreditation, and certification processes.

The Situation Center is an analytical system built on the principles and platform of business intelligence (BI), which allows in an operative mode to perform a detailed multi-vector analysis of the collected information with the possibility of an advanced in-depth analysis. "Laboratory information system"and "Medical images"information systems providing the collection, processing and transmission in EMZ of information about the diagnostic services rendered and their results.

The system for monitoring the sanitary and epidemiological situation is an information system that ensures the collection, generalization and analysis of information, timely informing the responsible departments of the Ministry of Health of the Republic of Kazakhstan and local authorities about the sanitary and epidemiological situation in all regions of the Republic. The patient's personal area provides patients with access to their own health data and access management for medical personnel, support of preventive PHC functions, self-monitoring of health status and notification of the need for health-related activities. Personal office of the doctor - provides the operative access of the doctor to the FTE and EMZ of his patients, a single entry point for the doctor in the information systems "Hospital "Polyclinic "First Aid".

Personal manager's office - a single point of access professionals MOH and its agencies, health departments, heads of health organizations to the functional information systems in accordance with its functional responsibilities.

Telemedicine - the provision of telemedicine services such as consultation, a consultation conference and videoselektornye meetings and meetings across the country, teleconferences, training seminars.

To ensure the functioning of the EPD as a platform for interaction in the health sector, additional intermediate software is needed - an integration platform for the exchange of medical data and the creation of a single information medical space. This platform consists of several components, the main of which are: DBMS and integration bus [7, 8].

Conceptual technical architecture

Technical architecture is the architecture of the hardware and software infrastructure that ensures the operation of application systems and the execution of operational (non-functional) requirements for the architecture of application systems and information.

Currently, the e-health architecture contains two types of information systems and applications. The first part is developed on the principle of client-server desktop applications. When using these systems,that all MOs have servers within the enterprise itself, and all users work with systems on the local network from their workstations. During the night, all the data that was updated during the day is sent from the local server to the top where the data is synchronized, and possible changes in the directories, data structures, and software codes are sent down. Exchange of operational information necessary for the process of medical care between centralized systems and nodes of UHMIS systems is carried out by accessing services through messaging through the transport environment. The second part of the applications is developed on the web technologies, in which users via web browsers access directly to the central data center of the Ministry of Health of the Republic of Kazakhstan, where web servers, applications and databases are located.

Figure 1 shows the architecture of software and hardware in the e-healthcare of the Republic of Kazakhstan.

As you can see in Figure 3, the architecture assumes the availability of a backup data center and a communication channel in order to ensure the continuity of the work of organizations in the event of a single data center failure or channel failure. At the beginning, the backup data center can be designed only for synchronous data replication to protect them in the event of data corruption (and disk) of the main data center. In this scenario, in the event of an accident, the data will be restored from the backup data center.

This architecture also shows that there are two categories of medical organizations: working in the cloud and working with a local server, and at the beginning most organizations will be from the second category. With the development of communication channels and the capacity of the data center, more and more enterprises will move from the second category to the first.

Inside the data center, all databases will be stored, among which the main ones are highlighted, which are shown in Figure 2.

It should be noted that the principle of operation of the database for FTE, and statistical data store differs because FTE database optimized for transaction (OLTP), and statistical data store is focused on the analytical work (OLAP) [6].
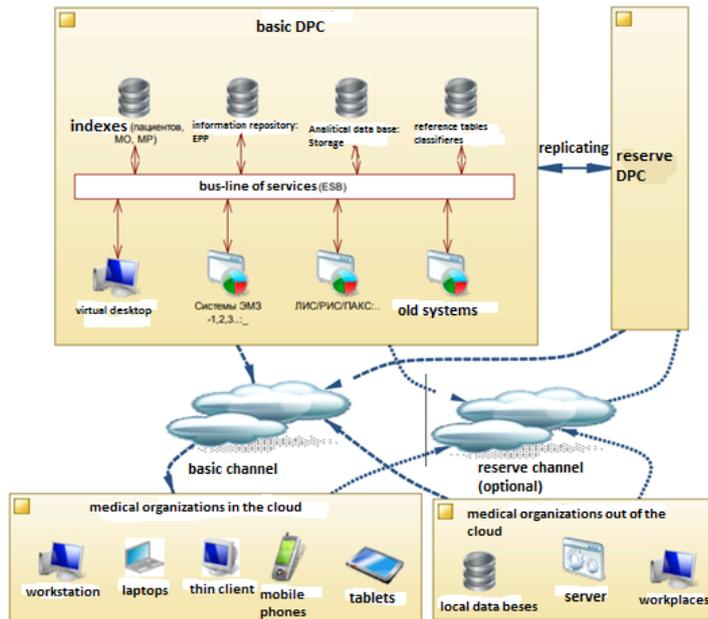
Figure 1. Architecture of software and hardware in e-healthcare.
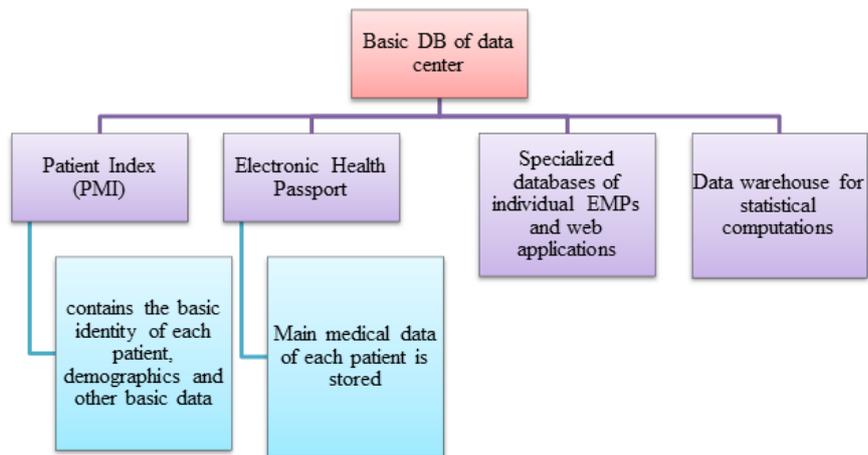
Figure 2. Main databases of e-Health Data Center.

Cloud computing in e-healthcare

The software and hardware intended for cloud computing will be presented in the form of the following three main layers of services:
    - Infrastructure as a service (IaaS);
    - Platform as a service (PaaS);
    - Software as a service (SaaS).
The architecture of cloud computing e-health is presented in Figure 3.

At the infrastructure level, e-health will provide services such as virtual servers, virtual desktops equipped with the required processing power.

At the Platform level, e-healthcare will provide, primarily, an integration bus that will allow the interconnection between different information systems working with different protocols. The bus will also provide access to various services published locally for use, in accordance with the SOA rules, all information systems. It is important to pay attention to the fact that the integration bus will allow to integrate various systems, including systems purchased directly by medical organizations [9, 10].
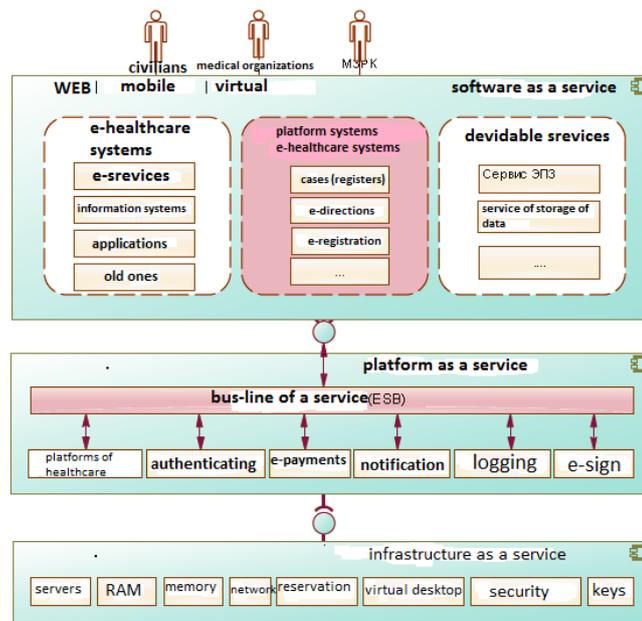
Figure 3. The architecture of cloud computing e-health.

Another important element of the e-Health Platform will be a database management system capable of storing both transactional FTE records and analytical repositories.

At the service level in the form of software (SaaS), several categories of software will be stored in the cloud:

1) Systems eHealth, which provide the basic functionality of business processes of health: e-government services in health care, information systems for individual health sectors, various types of web applications, legacy applications, operating separately or integrated via integration bus.

2) Applications originally designed to meet the requirements for a single information space, such as: new registers, new services;

3) Shared services are services intended for use by all systems - used in accordance with the principles of SOA. Examples of such services can be: data transfer to FTE, reading from FTE, EDS service, transfer of statistical reports to analytical storage, etc.

Conclusion

In conclusion, I would like to note that in order to build the correct architecture of IIAs, one must take into account: the development of ICT, incl. in health care is a strategic priority, and the need to emphasize the introduction of IC placed on the collection of analytical information for management and financial decisions, to the detriment of functionality and information, allowing healthcare professionals to provide safe, high-quality, timely and accessible health services.

References

1. The concept of e-health development of the Republic of Kazakhstan for 2013-2020. Order of the Ministry of Health of the Republic of Kazakhstan No. 498 of September 3, 2013
2. Message of the President of the Republic of Kazakhstan - the Leader of the Nation NA. Nazarbayev to the people of Kazakhstan, Astana, December 14, 2012
3. Uvalieva I.M., Belginova S.A. Informational and analytical system of the biochemical analysis of blood in children // International scientific conference "Applied Mathematics and Informatics"dedicated to the 25th anniversary of the Institute of Information and Computational Technologies. -Almaty. - 2016. - P. 8-19.
4. Zhuravlev EE et al. Interoperability in cloud computing // electronic resource] Journal of Radioelectronics (electronic journal). - 2013. - No. 9.
5. Kobrinsky BA, Kutukov VA Principles of construction and open questions of implementation of cloud technology in public health: The architecture of the information system // Information-measuring and control systems. - 2011. - T. 9. - No. 12. - P. 11-16.
6. Gimadeev, Sh.M. Integration of sources of medical information: goals and methodology / Sh.M. Gimadeev, A.I. Latypov, S.V. Radchenko // Doctor and Information Technologies. - 2006. - No. 6. - P.61-67.
7. Oleneva IV The current state of the problem of introduction of electronic medical cards in a single state information system // Medical alphabet. - 2011. - T. 4. - No. 20. - P. 8-10.

8. Information and communication technologies in the health care system of the Republic of Kazakhstan: problems and prospects for development A.R. Shopabaeva, R.M. Blatov, S.B. Sydykov, K.S. Zhakipbekov, K.M. Elshibekova Kazakh National Medical University. S.D. Asfendiyarova, Almaty, Republic of Kazakhstan.
9. Gribova V. V. et al. Cloud environment for the support of clinical medicine and education / / Doctor and information technology. - 2016. - No. 1.
10. Ismukhamedova AM, Uvalieva IM Analysis of decision support systems in medicine // Advanced Science. - 2017. - P. 99-101.

Time series in forecasting the volumes of air transportation in Kazakhstan

J. Jumabayeva, S. Burgumbayeva, and A. Iskakova

L.N. Gumilyov Eurasian National University, Astana, Kazakhstan
Jamilya_ast@mail.ru

Abstract In this paper, methods of mathematical statistics, namely, methods of regression and variance analysis construct and verify the quality of additive and multiplicative models of the time series predicting the volumes of air transportation in Kazakhstan. The methods of forecasting and planning the volume of passenger transportation are analyzed from the point of view of justifying the choice of the most expedient of them.

Keywords: forecasting, time series, additive model, multiplicative model, AirAstana

Introduction

The growth of air transportation around the world is one of the reliable indicators of economic growth. Improving the state of the world economy leads to an increase in the activity of the population of all countries and, ultimately, an increase in passenger flows. Air transport has always played and plays an important role in the economic and social development of all countries. Therefore, the availability of a developed ground infrastructure for civil aviation is a necessary condition for economic growth. And for the constant readiness of the infrastructure, it is important to determine the dependence of the future value on the past value within the process itself, and on the basis of this dependence make a forecast of its future value, formed as a result of their alignment and alignment.

The key task of the analytical department of the modern airline is the preliminary assessment and further qualitative forecast of the volume of own transportation and transportation of competitors. The pre-planned number of flights allows you to rationally distribute the company's resources without affecting the interests of the passenger at the same time. Such a policy leads to the exclusion of unprofitable "idle"flights and to an increase in the company's rating due to full compliance with the needs of customers.

Forecasting the volume of traffic is an integral part of the decision-making process; this is a systematic check of the company's resources, which allows to more fully use its advantages and to identify potential threats in a timely manner. The company must constantly monitor the dynamics of traffic volume

and alternative opportunities for the development of the air transportation market situation in order to best allocate available resources and choose the most appropriate areas of its activities.

The task of forecasting the volumes of passenger traffic is the basis for solving many problems of optimizing the air transportation system by the criteria directly related to the profitability index (revenues, costs, profits), since the optimality of plans obtained in solving these tasks depends primarily on the accuracy of forecasts.

The aim of the work is the development of decision-making models for air transport management, which, based on the analysis of the passenger's passenger statistics, will make forecasts on the size of passenger traffic, revenue and profitability. Thus, these models will enable the airline management to make decisions on the number of flights carried out, transport requests, ticket prices, etc.Time series models are mathematical prediction models that seek to determine the dependence of a future value on a past value within the process itself, and on the basis of this dependence make a forecast of its future value.

The main task in the construction of economic time series is the identification and statistical evaluation of the main trend of development of the studied process and deviations from it. The most common methods for analyzing time series are spectral, regression and correlation analysis, moving average and autoregressive models.

In this paper, methods of mathematical statistics, namely, methods of regression and variance analysis, construct and verify the quality of additive and multiplicative models of the time series predicting the volumes of air transportation in Kazakhstan.

There are data on the passengers of the company "AirAstana"for 2016-2017 (table 1) [1]-[2].

Table 1.

months	1	2	3	4	5	6	7	8
Number of passengers	405	351	422	402	462	578	669	730
months	9	10	11	12	13	14	15	16
Number of passengers	460	500	480	470	441	373	445	426
months	9	10	11	12	13	14	15	16
Number of passengers	451	531	646	680	530	510	460	490

It is necessary, with the use of these statistics, to construct additive and multiplicative time-series models.

The additive model

We proceed to construct an additive model that has the form [3]-[6]

$$Y = T + S + E. \tag{1}$$

Here, each level of the time series represents the sum of the trend (T), seasonal (S), and random (E) components.

1. We align the initial levels of the series using the moving average method. For these purposes:

1.1. Define the moving averages. The values found so aligned now do not contain the seasonal component.

1.2. Define the average of two successive moving averages - the centered moving averages, and then bring these values into line with the actual time points. The period of the seasonal period will be determined in 6 months, averaging will be performed for four months.

2. Using the difference between the actual levels of the series and the centered moving averages, we determine the estimates of the seasonal component and then apply these estimates in calculating the values of the seasonal component S. To this end, we calculate the average for each season of the seasonal component assessment S_i (Table 2). It is assumed that in models with a seasonal component, seasonal effects for the period are generally intertwined. This is expressed in the fact that in the additive model the sum of the values of the seasonal component over all periods is zero.

Table 2. Seasonal component data

Indicators	1	2	3	4	5	6
1	-	-	-14,67	-78,67	-81,83	27,83
2	102,5	160,5	-91,5	-13,5	26	18,5
3	1,833	-61,33	0,5	-52,67	-78,83	-13
4	88	120,5	-22,67	-	-	-
Total for the period	192,33	219,67	-128,33	-144,83	-134,67	33,33
The average estimate of the seasonal component	64,11	73,22	-32,08	-48,28	-44,89	11,11
Adjusted seasonal component S_i	60,25	69,36	-35,95	-52,14	-48,76	7,25

As a result, in this model we get:
64,111+73,222-32,083-48,278-44,889+11,111=23,194 The correction factor was: $k = 23,194/6 = 3,866$. Determine the corrected values for the seasonal component S_i and also input the data into the table.

3. Further exclude the impact of seasonal component by subtracting its value of each level of the original time series. Let us find the quantities $T + S = Y - E$.

These values are determined at each point in time and include a trend and a random component.

4. We calculate the component T of this model, for which it is necessary to perform analytical alignment of the series $(T + E)$ using the linear trend. We obtain the following results of analytical equalization:

$$T = 442,55 + 4,303t. \tag{2}$$

5. We find the values of the levels of the series obtained from the additive model. To do this, we add to the levels T the values of the seasonal component for the respective quarters. The average error of approximation was $A = 13,799\%$. Because less 20%, and thus the model is considered matched with good accuracy. Coefficient of determination $R^2 = 0,69$. Consequently, it can be said that the additive model accounts for 69% of the total variation in the levels of the time series. $F_H = 6,05$, which indicates the statistical significance of the equation.

6. Now let's make a prediction from the obtained model. The predicted value Y_t of the time series level in the additive model is the sum of the trend and seasonal components. Let's use the trend equation to determine the trend component: $T = 442,55 + 4,303t$.

The forecast for the first period is:
$T_{25} = 442,55 + 4,303 * 25 = 550,117$. The value of the seasonal component for the same period is: $S_1 = 60,245$. Thus, $Y_{25} = T_{25} + S_1 = 550,117 + 60,245 = 610,362$. This value will allow "AirAstana"Airlines to plan the number of passengers using their services for January 2018. Now forecast for the second period: $T_{26} = 442,55 + 4,303 * 26 = 554,419$. The value of the seasonal component for the same period is: $S_2 = 69,36$. Thus, $Y_{26} = T_{26} + S_2 = 550,117 + 69,36 = 619,477$. This value will allow AirAstana Airlines to plan the number of passengers using their services as of February 2018.

The multiplicative model

We now proceed to construct a multiplicative model, which has the form [3]-[6]

$$Y = T * S * E \tag{3}$$

According to this model, each level of the time series is represented as a product of trend (T), seasonal (S), and random (E) components.

1. Using the moving average method, we perform alignment of the initial levels of the series for what:
1.1. Define the moving averages. From the calculated aligned values, the seasonal component is now excluded.
1.2. In accordance with the actual time points, we give these values. For this purpose, we determine the average of two successive moving averages, the so-called centered moving averages.
2. Estimates of the seasonal component are defined as the quotient of dividing the actual levels of the series by the centered moving averages. When calculating

the seasonal component S, these estimates are used. For this purpose, we determine the average for each season of seasonal component assessment S_i (Table 3). Seasonal effects over the period necessarily mutually extinguished. This is expressed in the fact that in the multiplicative model the sum of the values of the seasonal component over all quarters is equal to the number of periods in the cycle. In our case, the number of periods of one cycle is 6.

Table 3. Seasonal component data

Indicators	1	2	3	4	5	6
1	-	-	-	0,88	0,9	1,06
2	1,2	1,29	0,82	0,94	0,99	1,04
3	0,99	0,85	1,01	0,92	0,89	0,99
4	1,17	1,22	0,95	-	-	-
Total for the period	3,36	3,36	2,79	2,74	2,79	3,08
Average seasonal component estimate	1,12	1,12	0,93	0,91	0,93	1,031
Adjusted seasonal component S_i	1,11	1,11	0,92	0,91	0,92	1,02

For our model, we have: $1,12 + 1,119 + 0,929 + 0,913 + 0,93 + 1,028 = 6,038$. Correction factor: $k = 6/6,038 = 0,994$.

Now calculate the adjusted values for the seasonal component S_i and add the data to the table.

3. Each level of the original series is divided into the corresponding values of the seasonal component. As a result, we get the values $T * E = Y/S$, which contain only a trend and a random component.

4. We calculate the component T of our model. To do this, we use the linear trend to perform an analytical alignment of the series $(T + E)$. The results of analytical equalization take the form:

$$T = 448,172 + 3,916t \tag{4}$$

Substituting in this equation the values $t = 1, \ldots, 24$, we find the levels T for each moment of time.

5. Multiplying the values T by the corresponding values of the seasonal component, we find the levels of the series. Coefficient of determination $R^2 = 0,71$. Thus, we can say that the multiplicative model explains 71% of the total variation in the levels of the time series. $F_H = 5,76$ which indicates the statistical significance of the equation.

6. Let's make a prediction on the constructed multiplicative model. The predicted value F_t of the time series level in the multiplicative model is calculated as the product of the trend and seasonal components.

To determine the trend component, let's use the trend equation (4).

Forecast for 1 period:

$T_{25} = 448,172 + 3,916 * 25 = 546,073$. The value of the seasonal component for the corresponding period is: $S_1 = 1,113$. Thus, $F_{25} = T_{25} * S_1 = 546,073 * 1,113 = 607,905$. This value will allow AirAstana Airlines to plan the number of passengers using their services for January 2018.

Forecast for the 2nd period: $T_{26} = 448,172 + 3,916 * 26 = 549,989$. The value of the seasonal component for the corresponding period is: $S_2 = 1,112$. Thus, $F_{26} = T_{26} * S_2 = 549,989 * 1,112 = 611,419$. This value will allow AirAstana Airlines to plan the number of passengers using their services for February 2018.

Conclusion

In the article, mathematical methods and algorithms for predicting time series have been developed and adapted taking into account the specifics of the tasks being solved in the interests of the airline "AirAstana and the creation of a system for forecasting the volumes of air transportation in Kazakhstan on their basis. Based on the forecast information obtained, the airline management will be able to increase the validity, objectivity and effectiveness of decision-making in business processes related to contracting and planning in forecasting the volume of air transportation in Kazakhstan, and also to use the company's resources most efficiently while not affecting the interests of passengers.

References

1. www.kΓrsΓж.kz.
2. Kazakhstan in 2017. Statistical yearbook. Astana (2018).
3. Noskov V.P. Econometrics, Vol. 1,2. "À business"RANH&GS, Moscow (2011).
4. Draper I., Smith G., Applied regression analysis, Vol. 1,2. Finance Statistics, Moscow (1987).
5. Ferster E., Rents B., Correlation and regression analysis methods. Finance Statistics. Moscow (1982).
6. Loskutov A.Y., Mikhailov A.S., Fundamentals of the theory of complex systems. Regular and chaotic dynamics. Moscow (2007)

Modeling of implants and process of dusting by the handling robot on implants by means of the virtual Roboguide simulator

I. B. Karymsakova[1], N. F. Denissova[1], and Y. V. Krak[2]

[1] D. Serikbayev East Kazakhstan State Technical University,
Ust-Kamenogorsk, Kazakhstan
[2] Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Abstract The modeling problem by means of the virtual Roboguide V6.40 simulator of system of creation of movements for drawing biocompatible coverings on implants on the basis of the handling Fanuc LR Mate 200 id robot is considered. For scanning of implant and creation of its spatial model the 3D SCAN 3D UNIVERSE scanner is used. For creation of program movements of the manipulator, taking into account rate of the movement, it is offered to use the Hermite cubic spline.

Keywords: implant, plasma dusting, virtual simulator of the robot, spline, Hermite's polynom.

Introduction

Creation of implants is rather knowledge-intensive and multi-stage process which consists not only in production of the substitute of human organs (generally bones), but also in development of additional technical and scientific solutions for a solution of the problem of compatibility of material of implant with human physiological system. The covering of surface of implants biocompatible material is one of ways of the solution of this problem. Use of industrial robots for dusting of implants biocompatible material is one of perspective decisions for which it is necessary to develop new methods and approaches. The purpose of this article is development of program control by the manipulator of the Fanuc LR Mate 200 id robot by means of the virtual Roboguide simulator taking into account rate of the movement for a solution of the problem of a covering of a surface of implants.

The object of research

The object of research are implants which are used for replacement in whole or in part of the injured human organs. As implants are implanted in a human body and they adjoin to other human organs, there is a problem of counteraction to a casting-off and contributions of survival of an implant which, at the modern level of development, is solved by means of a covering of implants biocompatible material [1-5].

Method of creation of the 3D model of an implant and creation of system of dusting

We will notice that one of effective ways of creation of implants is scanning of the damaged bone of the person and creation of its 3D model. We will use for this purpose the 3D SCAN 3D UNIVERSE scanner which restores the corresponding sizes, in the subsequent by these sizes in the Geomagic Design X program the implant 3D model is developed. The scan 3D UNIVERSE scanner represents the complex solution of hardware and the software for creation of 3D documentation of any objects and processing of results of measurements, for example for 3D - the press. The ability to integrate the scanner with the superfine software of Geomagic of the return engineering, allows to carry out automatic data processing in the parametrical CAD model and to exercise control of dimensional accuracy of preparation [6]. For creation of the program movement of the handling Fanuc LR Mate 200 id robot, the received 3D model of an implant is loaded into the virtual Roboguide V6.40 simulator. We will notice that the FANUC ROBOGUIDE application is run by modeling and simulation of movements of the handling robot, working off of teams for concrete preparation that, in turn, provides appreciable saving of time during creation of new settings of the movement. To reduce time for three-dimensional modeling, models of details can be imported from the personal computer in a CAD type of data. The larger library of the software for simulation allows users to choose and change details and the sizes. For carrying out an experiment of dusting the implant of a hip joint (fig. 1) was selected.
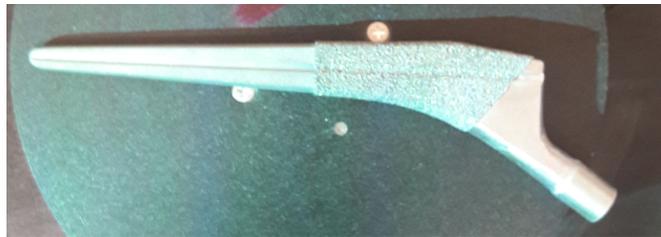
Figure 1. Implant of a hip joint

The implant is fixed in capture of the handling robot which has to move and rotate an implant for optimum dusting on a surface. High-quality drawing a covering requires performance of the following conditions:

1. the plasmatron sputters metal powder with laminar plasma stream, there are at distance about 150 mm from an object at an angle 90°.
2. diameter of dusting has to be − 10-25 mm (we will notice that the poristy there will be a surface, the, according to the conclusions of experts, better the implant gets accustomed in a human body), thickness of covering 0,05...1,0 of mm.

3. admissible losses of powder of metal when dusting have to make 10-15%.
4. used material − VT6 titanium.

Creation of system of the movement of the handling robot

For drawing up mathematical model of the movement we will use the following assumptions. Let at a stage of planning of a trajectory of the movement some final sequence of points − values of the generalized coordinates be received. We will notice that modern racks of control of industrial robots allow to find and write down these values. It is required to construct on this sequence dependence, continuous on time, which in discrete timepoints coincides with preset values. In other words − it is necessary to construct an interpolation polynom which in certain known timepoints will coincide with values in knots. For a solution of the problem of effective drawing a biocompatible covering it is necessary to demand that the first derivative of this interpolation polynom in knots of a grid coincided with preset values. From the technical point of view it means that passing through the set knots and with the set rate in these knots is required. For the solution of such task it is possible to use cubic splines − algebraic polynoms of the third degree with constant coefficients on each of intervals. In case of performance of conditions of passing through the set points and with the set rate the Hermite cubic spline by which the trajectory of the movement of the robot is determined [7-9] is under construction. We will survey the solution of this task in the environment of Matlab. In the environment of Matlab there are built-in functions for creation of polynoms, including Hermite's polynom, also functions for finding of coefficients of a polynom [10]. We will simulate creation of a trajectory of the movement on the offered scheme. Function which finds pp-representation for a cubic Hermite interpolant in Matlab can look as follows:

```
function p=cubic(x,y,d);
n=length(x);
a=zeros(n-1,4);
for i=1:n-1
a(i,1)=2*y(i)-2*y(i+1)+d(i)+d(i+1);
a(i,2)=-3*y(i)+3*y(i+1)-2*d(i)-d(i+1);
a(i,3)=d(i);
a(i,4)=y(i);
```
$$a(i,1) = \frac{a(i,1)}{(x(i+1)-x(i))^3};$$
$$a(i,2) = \frac{a(i,2)}{(x(i+1)-x(i))^2};$$
$$a(i,3) = \frac{a(i,3)}{(x(i+1)-x(i))^1};$$
$$a(i,4) = a(i,4);$$
```
end;
```
Model test data for use of this function:
```
x=[0 30 60 90 120 150 200 220];
y=[650 675 710 715 725 730 735 740];
d1=[10 10 10 10 10 10 10 10];
```

```
p1=cubic(x,y,d1);
xx=0:.10:220;
yy1=ppval(p1,xx);
plot(x,y,xx,yy1);
axis([0 250 600 750]);
title('Hermitian cubic spline')
xlabel('x'); ylabel('y');
legend('spline', 'Hermitian cubic spline')
```
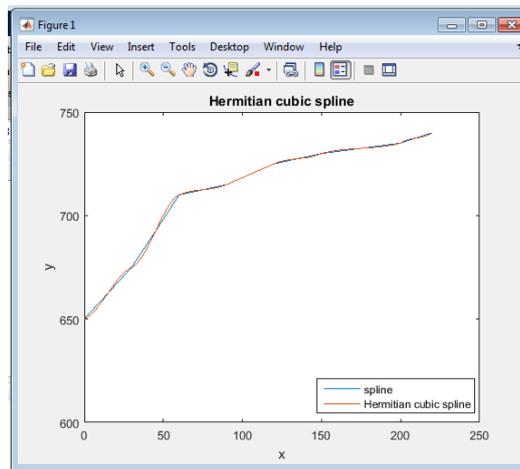

Figure 2. Hermite cubic interpolation with an obvious task of derivatives

As appears from fig. 2, when using the Hermite polynom the requirement of smoothness of function is fulfilled, the passing condition on the set trajectory with the set rates is respectively satisfied though generally, the Hermite cubic spline doesn't meet requirements of continuity of the second derivative in grid knots. The research of influence of this restriction for quality of trajectories of the movement when dusting demands additional researches. Realization of movements on the virtual Roboguide V6.40 simulator In the virtual Roboguide V6.40 simulator the purposes are set, the group of the purposes and on them is under construction the program for the movement of the handling robot for the set coordinates and rates is generated. The screenshot of work of the program on the simulator is given in fig. 3.

Conclusion

On the example of the implant of hip joint simulation of the movement of the robot by means of the virtual Roboguide V6.40 simulator was constructed, for creation of an interpolation polynom with the set coordinates and rates the Hermite spline was used. Test examples showed feasibility and efficiency of the offered approach.
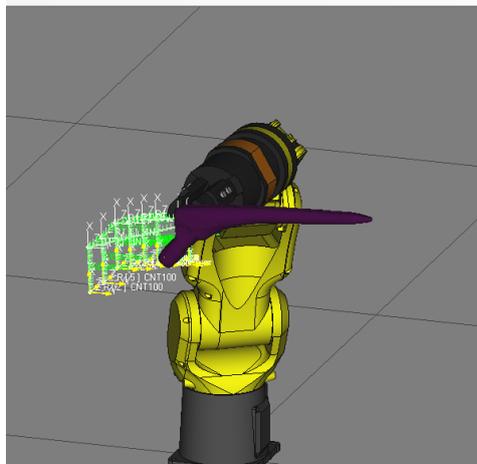
Figure 3. Generation of TR of the program in the virtual Roboguide V6.40 simulator

References

1. http://irobs.ru/robotyi-dlya-napyileniya1
2. http://studall.org/all-109356.html
3. http://metcompozit.ru/tech.html
4. Alontseva D., Krasavin A., Kolesnikova T., Russakova, A. Modeling of Processes Taking Place during Powder Coating Treatment by an Electron Beam or a Plasma Jet// ActaPhysicaPolonica A Vol. 125, No. 6. - P. 1275-1279 (2014)
5. Dostanko A.P., Grushetski S.V., Kiselevski L.I., Pikul M.I., Shiripov V.Y. Plasmennaya metalizatsya v vakuume. – Nauka I technika (1983)
6. http://on-v.com.ua/novosti/texnologii-i-nauka/universalnyj-3d-skaner-scan3d-universe/
7. Kirichenko N.F., Soroka R.A., KrakY.V. «Manipulacionnye roboty. Algoritmicheskoe I programmnoe obespechenie sredstv upravlenya dvijeniem», Uchebnoe posobie, Kiev KGU (1987)
8. Popov E.P., Verechagin A.F., Zenkevich S.L. Manipulacionnye roboty: dinamika i algoritmy. M., Nauka (1978)
9. Zavyalov Y.S., Kvasov B.I., Mirochnichenko V.L. Metody splain-funcii. M., Nauka (1980)
10. N.Y.Zolotyh. Matlab v nauchnyh issledovanyah. Nizhegorodsk (2004)

The modification of fictitious domain method for the model of fluid

A. Krykpayeva[1] and Zh. Baitulenov[2]

[1] D. Serikbayev East Kazakhstan State Technical University,
Ust-Kamenogorsk, Kazakhstan,
[2] Al-Farabi Kazakh National University, Almaty, Kazakhstan
krykpaeva64@mail.ru

Abstract This paper researches one of the adaptations of fictititous domain method for the stationary model of non-Newtonian fluid. The existence and convergence of generalized solutions auxiliary problem is being proven. The assessment of the convergence of approximate problem's strong solution is outlined.

Keywords: generalized solution, fictitious domain, prior estimation, non-Newtonian fluid, convergence.

Introduction and Problem

The Method of fictitious domain is often used to solve wide spectrum of mathematical problems. Good introduction to this method is given in the work [1]. The method can also be used to estimate the solution's convergence rate. Therefore, the investigation of the new applications of the fictitious domain method that improve the solution's convergence rate estimation is actual. In the work [2] the another modification of the method was suggested to the Dirichlet problem for Poisson equation. In that work [2] the better estimation of convergence was achieved in comparison to known types of the fictitious domain method. In the present paper, the new modification is described for the first time. This modification is applied to the stationary non-linear model of Oldroyd type non-Newtonian fluids.

So, we consider a stationary mathematical model on non-Newtonian liquid in the bounded domain $\Omega \subset R^3$ [3]:

$$Re\,(v \cdot \nabla)\,v + \nabla P = (1 - \alpha)\,\Delta v + \nabla \cdot S + f, \qquad (1)$$

$$S + We\,(v \cdot \nabla)\,S = 2\alpha D, \qquad (2)$$

$$divv = 0, \qquad (3)$$

with boundary conditions

$$v|_G = 0, \qquad (4)$$

where: $x = (x_1, x_2, x_3) \in \Omega$, $v = (v_1(x), v_2(x), v_3(x))$ - fluid velocity vector, $P = P(x)$- pressure, $S = S(x)$- elastic part of stress tensor, tensor function

$D = \left(\nabla v + (\nabla v)^T \right)/2$ - deformation velocities tensor, D, S - second rank tensors with components $\frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$, S_{ij}; $Re = \frac{uL}{\mu}$, $We = \frac{\lambda_1 u}{L}$ - Reynolds and Weisenberg numbers, $\alpha = 1 - \frac{\lambda_2}{\lambda_1}$ - numeric parameter, λ_1- relaxation time, λ_2- lagging time, $0 < \lambda_2 < \lambda_1$. u, L- characteristic velocity and model dimension, $f = (f_1(x), f_2(x), f_3(x))$ - mass forces vector, G- boundary of domain Ω.

In the work [4] the classical type of fictitious domain method is substantiated for problems (1)-(4).

The following modification of the method of fictitious domain is investigated with prolongation by leading coefficients in the $D_0 \supset \Omega$:

$$Re \, (v^\varepsilon \cdot \nabla) \, v^\varepsilon + \nabla P^\varepsilon = div \, (K \, (v^\varepsilon) \, \nabla v^\varepsilon) + \nabla \cdot S^\varepsilon + f^\varepsilon, \tag{5}$$

$$S^\varepsilon + We \, (v^\varepsilon \cdot \nabla) \, S^\varepsilon = 2\alpha D^\varepsilon, \tag{6}$$

$$div v^\varepsilon = 0, \tag{7}$$

$$v^\varepsilon \, |_{G_1} = 0, \tag{8}$$

with conditions connection

$$[v^\varepsilon]|_G = 0, \quad \left[K \, (v^\varepsilon) \, \frac{\partial v^\varepsilon}{\partial n} - P^\varepsilon n + S^\varepsilon \cdot n \right]\Big|_G = 0, \quad [S^\varepsilon]|_G = 0, \tag{9}$$

where: $0 < \beta < 1$, G_1 - boundary of D_0, $G_1 \bigcap G = \emptyset$, $\varepsilon > 0$, n - normal vector to G_1,

$$\xi(x) = \begin{cases} 0, & x \in \Omega \\ 1, & x \in; D_1 = D_0 \backslash \Omega \end{cases}. \quad f^\varepsilon(x) = \begin{cases} f(x), & x \in \Omega \\ 0, & x \in; D_1 \end{cases}$$

$$K^\varepsilon(v^\varepsilon) = \begin{cases} 1 - \alpha, & when \quad x \in \Omega, \\ \frac{1-\alpha}{\varepsilon \|\nabla v^\varepsilon\|_{L_2(D_1)}^\beta}, & 0 < \beta < 1, \quad when \quad x \in D_1 = D_0/\Omega, \end{cases}$$

If $\beta = 0$ in the problem (5)-(9), then this problem becomes the variant of traditional fictitious domain method [4]. The fields $J(D_0)$ and $J^1(D_0)$ - are closures in normals $L_2(D_0)$ and $W_2^1(D_0)$ of finite fields in D_0 indefinitely differentiating solenoid vector-functions.

Definition 1.1. *The weak solution of problems (5)-(9) is functions* $v^\varepsilon \in J^1(D_0)$, $S^\varepsilon \in L_2(D_0)$ *satisfy to the following:*

$$-Re \, ((v^\varepsilon \cdot \nabla) \, \Phi v^\varepsilon)_{L_2(D_0)} + (1 - \alpha) \, (\nabla v^\varepsilon, \nabla \Phi)_{L_2(\Omega)} =$$

$$= - \, (S^\varepsilon : \nabla \Phi)_{L_2(D_0)} - \frac{1 - \alpha}{\varepsilon \, \|\nabla v^\varepsilon\|_{L_2(D_1)}^\beta} \, (\nabla v^\varepsilon, \nabla \Phi)_{L_2(D_1)} + (f^\varepsilon, \Phi)_{L_2(D_0)}, \tag{10}$$

$$(S^\varepsilon : \varphi)_{L_2(D_0)} - We \, ((S^\varepsilon \cdot \nabla) \, \varphi, v^\varepsilon)_{L_2(D_0)} = 2\alpha \, (D^\varepsilon : \varphi)_{L_2(D_0)} \tag{11}$$

$$\forall \Phi \in J^1(D_0), \quad \varphi \in W_2^1(D_0).$$

Here:

$\int\limits_{D_0} (S : \varphi)\, dx = \sum\limits_{i,j=1}^{3} \int\limits_{D_0} S_{ij} \cdot \varphi_{ij} dx$ - tensor accumulation $L_2 (D_0)$,

$(u,v)_{L_2(D_0)} = \int\limits_{D_0} u \cdot v dx$ - scalar vector multiplication in $L_2 (D_0)$.

Definition 1.2. *The strong solution of (5)-(9) is given by functions $v^\varepsilon (x)$, $S^\varepsilon (x)$, $P^\varepsilon (x)$, that have quadratic summable derivatives included in equations (5)-(6) and that satisfy (5)-(9) almost everywhere in D_0.*

The existence and converge of solutions

In the following chapter the description and proof of main theorems are given.

Theorem 2.1. *Lets $f \in L_{\frac{6}{5}} (\Omega)$. Then there is at least one generalized decision for the problem (5)-(9) and the following estimation for the solution is valid:*

$$\|v_x^\varepsilon\|_{L_2(\Omega)}^2 + \|S^\varepsilon\|_{L_2(D_0)}^2 + \frac{1}{\varepsilon} \|v_x^\varepsilon\|_{L_2(D_1)}^{2-\beta} \le C < \infty.$$

From now on, let's use C to denote all the constants that depend only on problem's data and different constants from known inequalities [5], and not depend on the looked for functions and the parameter ε.

Proof. Approximated solution of problem (5)-(9) will be found with $v_N^\varepsilon (x) = \sum\limits_{j=1}^{N} \alpha_j \omega_j$, where $\{\omega_j\}$ - basis in $J^1 (D_0)$. α_j can be found from:

$$-Re\left((v_N^\varepsilon \cdot \nabla) \omega_j, v_N^\varepsilon\right)_{L_2(D_0)} + (1-\alpha)\left(\nabla v_N^\varepsilon, \nabla \omega_j\right)_{L_2(\Omega)} =$$

$$= -\left(S_N^\varepsilon : \nabla \omega_j\right)_{L_2(D_0)} - \frac{1-\alpha}{\varepsilon \left\|\nabla v_N^\varepsilon\right\|_{L_2(D_1)}^\beta} \left(\nabla v_N^\varepsilon, \nabla \omega_j\right)_{L_2(D_1)} + \qquad (12)$$

$$+ \left(f^\varepsilon, \omega_j\right)_{L_2(D_0)},$$

$$S_N^\varepsilon + We\left(v_N^\varepsilon \cdot \nabla\right) S_N^\varepsilon = 2\alpha D_N^\varepsilon, \quad j = 1, 2, ..., N. \qquad (13)$$

To proof the existence of solutions lets regularize problem (12)-(13):

$$-Re\left((v_N^{\nu,\varepsilon} \cdot \nabla) \omega_j, v_N^{\nu,\varepsilon}\right)_{L_2(D_0)} + (1-\alpha)\left(\nabla v_N^{\nu,\varepsilon}, \nabla \omega_j\right)_{L_2(\Omega)} =$$

$$= -\left(S_N^{\nu,\varepsilon} : \nabla \omega_j\right)_{L_2(D_0)} - \frac{1-\alpha}{\varepsilon \left\|\nabla v_N^{\nu,\varepsilon}\right\|_{L_2(D_1)}^\beta} \left(\nabla v_N^{\nu,\varepsilon}, \nabla \omega_j\right)_{L_2(D_1)} + \qquad (14)$$

$$+ \left(f^\varepsilon, \omega_j\right)_{L_2(D_0)},$$

$$S_N^{\nu,\varepsilon} + We\left(v_N^{\nu,\varepsilon} \cdot \nabla\right) S_N^{\nu,\varepsilon} = 2\alpha D_N^{\nu,\varepsilon} + \nu \Delta S_N^{\nu,\varepsilon},$$

with boundary conditions

$$\left.\frac{\partial S_N^{\nu,\varepsilon}}{\partial n}\right|_{G_1} = 0. \qquad (15)$$

Working also as in [3], we can find the estimate for the problem (14)-(15):

$$\|\nabla v_N^{\nu,\varepsilon}\|_{L_2(\Omega)}^2 + \|S_N^{\nu,\varepsilon}\|_{L_2(D_0)}^2 + \nu \|\nabla S_N^{\nu,\varepsilon}\|_{L_2(D_0)}^2 + \frac{1}{\varepsilon} \|v_N^{\nu,\varepsilon}\|_{J^1(D_1)}^{2-\beta} \leq$$

$$\leq C \|f\|_{L_{\frac{6}{5}}(\Omega)}^2. \tag{16}$$

Following the path taken in [3] and using estimate (16) and Brower lemma, the existence theorem for regularized problem (14)-(15) can be proven.

The constant C in (16) does not depend on ε, ν. Therefore, the subsequences can be derived from sequences $\{v_N^{\nu,\varepsilon}\}, \{S_N^{\nu,\varepsilon}\}$ that satisfy $\nu \rightarrow 0$ and have following properties:

$v_N^{\nu,\varepsilon} \rightarrow v_N^{\varepsilon}$ weak in $J^1(D_0)$, $v_N^{\nu,\varepsilon} \rightarrow v_N^{\varepsilon}$ strong in $L_4(D_0)$,
$S_N^{\nu} \rightarrow S_N^{\varepsilon}$ weak in $L_2(D_0)$, $\sqrt{\nu} S_N^{\nu} \rightarrow 0$ weak in $W_2^1(D_0)$.

In (14) let's multiply second equation by $\phi \in W_2^1(D_0)$ and integrate it in D_0 using the chosen sequences limit, resulting in:

$$-Re\left((v_N^{\varepsilon} \cdot \nabla)\omega_j, v_N^{\varepsilon}\right)_{L_2(D_0)} + (1-\alpha)\left(\nabla v_N^{\varepsilon}, \nabla \omega_j\right)_{L_2(\Omega)} =$$
$$= -\left(S_N^{\varepsilon} : \nabla \omega_j\right)_{L_2(D_0)} - \frac{1-\alpha}{\varepsilon\|\nabla v_N^{\varepsilon}\|_{L_2(D_1)}^{\beta}}\left(\nabla v_N^{\varepsilon}, \nabla \omega_j\right)_{L_2(D_1)} + (f^{\varepsilon}, \omega_j)_{L_2(D_0)},$$

$$\left(S_N^{\varepsilon} : \varphi\right)_{L_2(D_0)} - We\left((S_N^{\varepsilon} \cdot \nabla)\varphi, v_N^{\varepsilon}\right)_{L_2(D_0)} = 2\alpha\left(D_N^{\varepsilon} : \varphi\right)_{L_2(D_0)}.$$

From here we have (12)-(13). After we multiply (12),(13) by ξ_j, S_N^{ε} respectively, then integrate in D_0:

$$(1-\alpha)\|\nabla v_N^{\varepsilon}\|_{L_2(\Omega)}^2 + \frac{1}{\varepsilon}\|\nabla v_N^{\varepsilon}\|_{L_2(D_1)}^{2-\beta} = -\left(S_N^{\varepsilon} : \nabla v_N^{\varepsilon}\right)_{L_2(D_0)} + (f^{\varepsilon}, v_N^{\varepsilon})_{L_2(D_0)},$$
$$\|S_N^{\varepsilon}\|^2 = 2\alpha\left(D_N^{\varepsilon} : S_N^{\varepsilon}\right)_{L_2(D_0)}.$$

Using the $S_N^{\varepsilon} : \nabla v_N^{\varepsilon} = S_N^{\varepsilon} : D_N^{\varepsilon}$, the following estimate can easily be found:

$$\|\nabla v_N^{\varepsilon}\|_{L_2(\Omega)}^2 + \|S_N^{\varepsilon}\|_{L_2(D_0)}^2 + \frac{1}{\varepsilon}\|v_N^{\varepsilon}\|_{J^1(D_1)}^{2-\beta} \leq C. \tag{17}$$

In estimate (17) constant C does not depend on N, ε. Therefore, the subsequences with following properties can be defined from sequences $\{v_N^{\varepsilon}\}, \{S_N^{\varepsilon}\}$:

$v_N^{\varepsilon} \rightarrow v^{\varepsilon}$ weak in $J^1(D_0)$, $v_N^{\varepsilon} \rightarrow v^{\varepsilon}$ strong in $L_4(D_0)$,
$S_N^{\varepsilon} \rightarrow S^{\varepsilon}$ weak in $L_2(D_0)$,

when $N \rightarrow \infty$, specifically for $v^{\varepsilon}, S^{\varepsilon}$ the following estimate can take place:

$$\|\nabla v^{\varepsilon}\|_{L_2(\Omega)}^2 + \|S^{\varepsilon}\|_{L_2(D_0)}^2 + \frac{1}{\varepsilon}\|\nabla v^{\varepsilon}\|_{L_2(D_1)}^{2-\beta} \leq C. \tag{18}$$

Thus, theorem 2.1 is proved.

Theorem 2.2. *Assume all conditions of the theorem 2.1 are carried out. Then when $\varepsilon \to 0$ the solution of the problem (5)-(9) converges to the solution of the problem (1)-(4).*

Similar to theorem 2.1, this theorem can be proved on basis of estimates (18).

Let's find estimate of convergence rate of strong solution of the problem (5)-(9) to the strong solution of problem (1)-(4). Assume that the next condition is carried out.

Condition 2.1. *Let's suppose there are strong solution of problem (5)-(9) and the problem (1)-(4) with estimate:*

$$\exists C_1, \ \infty > C_1 > 0 : \quad \|v\|_{L_3(\Omega)} + \ \|\nabla S\|_{L_3(\Omega)} \le C_1, \tag{19}$$

numbers Re and We satisfy the following inequality

$$\frac{\sqrt[3]{6}}{\alpha}\,(C_1\,We)^2 - 2\,\sqrt[6]{48}\,C_1 Re < 1 - \alpha. \tag{20}$$

Theorem 2.3. *If all conditions of Theorem 2.1 and Condition 2.1 are satisfied than the following estimate is true:*

$$\|v^{\varepsilon} - v\|^2_{W_2^1(\Omega)} + \|S^{\varepsilon} - S\|^2_{L_2(\Omega)} \le C\varepsilon^{\frac{1}{1-\beta}}, 0 < \beta < 1. \tag{21}$$

Proof. Scalar multiply (1) by $\psi \in J^1(D_0)$ in $L_2(\Omega)$:

$$-Re\int\limits_{\Omega} ((v \cdot \nabla)\,\psi v)\,dx + (1-\alpha)\int\limits_{\Omega} \nabla v \cdot \nabla\psi dx - (1-\alpha)\int\limits_{G} \frac{\partial v}{\partial n}\psi dG +$$

$$+ \int\limits_{\Omega} S : \nabla\psi dx - \int\limits_{G} S \cdot n \cdot \psi dG + \int\limits_{G} P \cdot n \cdot \psi dG - \int\limits_{\Omega} f\psi dx = 0. \tag{22}$$

Scalar multiply (5) by ψ in $L_2(D_0)$:

$$-Re\int\limits_{D_0} ((v^{\varepsilon} \cdot \nabla)\,\psi v^{\varepsilon})\,dx + (1-\alpha)\int\limits_{\Omega} \nabla v^{\varepsilon} \cdot \nabla\psi dx + \int\limits_{D_0} S^{\varepsilon} : \nabla\psi dx -$$

$$- \int\limits_{D_0} f^{\varepsilon}\psi dx + \frac{1-\alpha}{\varepsilon\,\|\nabla v^{\varepsilon}\|^{\beta}_{L_2(D_1)}}\int\limits_{D_1} \nabla v^{\varepsilon}\nabla\psi dx = 0. \tag{23}$$

Continue $v(x)$ and $S(x)$ outside Ω using 0. Define $\omega = v^{\varepsilon} - v$, $\theta = S^{\varepsilon} - S$, $\psi = \omega$ and consider differences (23) and (22):

$$-Re\int\limits_{D_0} ((\omega \cdot \nabla)\,\omega v^{\varepsilon})\,dx + (1-\alpha)\int\limits_{\Omega} \nabla\omega \cdot \nabla\omega dx + \int\limits_{D_0} \theta \cdot \nabla\omega dx -$$

$$- \int\limits_{G} P \cdot n \cdot \omega\,dG + (1-\alpha)\int\limits_{G} \frac{\partial v}{\partial n}\omega dG + \int\limits_{G} S \cdot n \cdot \omega dG + \frac{1}{\varepsilon}\,\|\nabla\omega\|^{2-\beta}_{L_2(D_1)} = 0. \tag{24}$$

Scalar multiply (6) by ϕ in $L_2(D_0)$:

$$\int\limits_{D_0} S^\varepsilon : \phi dx - We \int\limits_{D_0} (v^\varepsilon \cdot \nabla)\,\phi \cdot S^\varepsilon dx = 2\alpha \int\limits_{D_0} D^\varepsilon : \phi dx, \qquad (25)$$

Scalar multiply (2) by ϕ in $L_2(\Omega)$:

$$\int\limits_{\Omega} S \cdot \phi dx - We \int\limits_{\Omega} (v \cdot \nabla)\,\phi \cdot S dx = 2\alpha \int\limits_{\Omega} D \cdot \phi dx, \qquad (26)$$

Let's $\phi = \theta$, then from (25) and (26) derive:

$$\|\theta\|^2 - We \int\limits_{D_0} (\omega \cdot \nabla)\,\theta \cdot S dx = 2\alpha \int\limits_{D_0} (D^\varepsilon - D) \cdot \theta dx. \qquad (27)$$

Divide (27) by 2α and add with (24):

$$(1-\alpha)\,\|\nabla\omega\|^2_{L_2(\Omega)} + \frac{1}{2\alpha}\,\|\theta\|^2_{L_2(D_0)} + \frac{1}{\varepsilon}\,\|\nabla\omega\|^{2-\beta}_{L_2(D_1)} - \frac{We}{2\alpha} \int\limits_{D_0} (\omega \cdot \nabla)\,\theta \cdot S dx =$$

$$= Re \int\limits_{D_0} ((\omega \cdot \nabla)\,\omega v^\varepsilon)\,dx + (1-\alpha) \int\limits_{G} \frac{\partial v}{\partial n} \omega dG + \int\limits_{G} S \cdot n \cdot \omega dG - \int\limits_{G} P \cdot n \cdot \omega dG. \quad (28)$$

Let's estimate integrals in (28) by using Gelder inequalities embedding theorem of Jung and (19):

$$\frac{We}{2\alpha} \int\limits_{D_0} (\omega \cdot \nabla)\,\theta \cdot S dx = -\frac{We}{2\alpha} \int\limits_{D_0} (\omega \cdot \nabla)\,S \cdot \theta dx \le \frac{We}{2\alpha}\,\|\nabla S\|_{L_3}\,\|\theta\|_{L_2}\,\|\omega\|_{L_6} \le$$

$$\le \frac{We}{2\alpha} C_1\,\|\theta\|_{L_2}\,\sqrt[6]{48}\,\|\omega_x\|_{L_2} \le \frac{1}{4\alpha}\,\|\theta\|^2 + \alpha\left(\frac{We}{2\alpha} C_1\right)^2 \sqrt[3]{48}\,\|\omega_x\|^2,$$

$$Re \int\limits_{D_0} ((\omega \cdot \nabla)\,\omega v^\varepsilon)\,dx = Re \int\limits_{D_0} (\omega \cdot \nabla)\,\omega v\,dx \le Re\,\|v\|_{L_3}\,\|\omega_x\|_{L_2}\,\|\omega\|_{L_6} \le$$

$$\le C_1 Re \sqrt[6]{48}\,\|\omega_x\|^2,$$

$$\int\limits_{G} \left((1-\alpha)\frac{\partial v}{\partial n} + Sn - Pn\right) \omega dG \le \left(\left\|\frac{\partial v}{\partial n}\right\|_{L_2(G)} + \|S\|_{L_2(G)} + \|P\|_{L_2(G)}\right) \times$$

$$\times \|\omega\|_{L_2(G)} \le C\,\|\omega\|_{L_2(G)} \le C\,\|\nabla\omega\|_{L_2(D_1)} = C\varepsilon^{\frac{-1}{2-\beta}}\,\|\nabla\omega\|_{L_2(D_1)}\,\varepsilon^{\frac{1}{2-\beta}} \le$$

$$\le \delta\varepsilon^{-1}\,\|\omega_x\|^{2-\beta}_{L_2(D_1)} + \frac{C}{\delta}\varepsilon^{\frac{1}{1-\beta}},$$

where Jung inequalities are used with $p = 2 - \beta$ and $q = \frac{2-\beta}{1-\beta}$.

As a result we receive from (28) when $\delta = \frac{1-\alpha}{2}$:

$$
\left(\frac{1-\alpha}{2} - \alpha \left(\frac{We}{2\alpha} C_1 \right)^2 \sqrt[3]{48} - C_1 Re \sqrt[6]{48} \right) \|\omega_x\|_{L_2(\Omega)}^2 + \frac{1}{2\alpha} \|\theta\|_{L_2(D_0)}^2 +
$$
$$
+ \frac{1-\alpha}{2\varepsilon} \|\omega_x\|_{L_2(D_1)}^{2-\beta} \leq C\varepsilon^{\frac{1}{1-\beta}}
\tag{29}
$$

Thus the next estimation follows from (29) by using (20):

$$
\|\omega_x\|_{L_2(\Omega)}^2 + \frac{1}{\varepsilon} \|\omega_x\|_{L_2(D_1)}^{2-\beta} + \|\theta\|_{L_2(D_0)}^2 \leq C\varepsilon^{\frac{1}{1-\beta}} \rightarrow C\varepsilon^\infty, when \ \beta \rightarrow 1.
$$

Another words, we receive the estimation of convergence rate of (21). Thus the theorem 2.3 is proved.

References

1. Vabishchevich P.N. The Method of Fictitious Domains in Problems of Mathematical Physics. Izdatelstvo Moskovskogo Universiteta, Moskva, 1991. p. 156 (in russian).
2. Kuttykozhayeva Sh., Danaev N., Smagulov S., Orunkhanov M. Superconvergence in the New Version Fictitious Domains Method // Abstracts of the First International Conference "Inverse Problems: Modelinq and Simulation" held on july 14-21. - Turkey, Fethiye, 2002. - p. 52-56.
3. Turganbaev E. M. Initial-boundary value problems for the equations of a viscoelastic fluid of Oldroyd type. Siberian Mathematical Journal, 1995, 36:2, p. 389-403.
4. Krykpayeva A.A. Method of fictitious domains for equations of non-Newtonian fluids. - Vestnik KazNU, N5 (19), Almaty, 2000, - p.92-97. (in russian).
5. Antontsev, S. N.; Kazhikhov, A. V.; Monakhov, V. N., Boundary Value Problems in Mechanics of Nonhomogeneous Fluids. Amsterdam etc., North Holland 1990. XII, p. 309.

Organization of work with binary data in the system of distance learning (on the example of the distance learning system at D. Serikbayev EKSTU)

S. K. Kumargazhanova, E.M. Fedkin, and Zh. T. Konurbayeva

D. Serikbayev EKSTU, Oskemen, Republic of Kazakhstan
skumargazhanova@gmail.com,EFedkin@mail.ru

Abstract At the modern stage of development the majority part of the information is stored in files of different types. Use files to store is very convenient for users and the distance learning system have to provide capability for the storing and the transfer of the files. By working with the binary data needs to decide of some questions: where storing files and how exclude duplication files. In the distance learning system of EKSTU the binary data is stored in the database by using FileStream technology, and for excluding dublication files used to cryptographic hash-function of algorithm SHA-1.

Keywords: distance learning, binary data, education portal, hash-function, SHA-1

Introduction

At the present stage of development, most of the information is stored in electronic format in the form of files having a different data storage format. Most of these files have a binary storage format, i.e. The data is represented as a sequence of bytes. Work with such files is carried out by the corresponding program or software package. For example, the most common format for storing text documents is the Microsoft Word format, whose files have such extensions as ".doc ".docx"and a number of others. The use of files to store various information is very convenient for users, as it allows them to store and exchange information.

Since there is a large amount of information resources in the form of files, the distance learning system should provide the ability to store and transfer data stored in files for its users.

Issues related to storing binary data

When organizing the storage of binary data in the distance learning system, a number of issues arise that need to be addressed.

The first of such questions is the question of where to store files. Possible solutions to this issue can be:

1. Files can be stored directly in the database if the appropriate database management system supports this storage. With this method of storage, files will be stored centrally, which simplifies access to them. It also has the advantage that database management systems have a backup system that allows you to transfer files together with the database. However, there are a number of problems. First, if the number of files is large, this will lead to a dramatic increase in the size of the database. Secondly, to write and read files, you need a permanent connection to the database when performing these operations, which can lead to performance degradation.

2. Files can be stored in the file system. In a database with this type of storage, you only need to store only the location of the corresponding files in the file system. With this method of storage, the interaction time with the database will be drastically reduced, and the file system of the operating system will be used to work with files, which dramatically increases the system's performance. The main drawback of this system is that the data is stored in different systems, which complicates the maintenance of data integrity. The procedure of reservation and data transfer is also complicated. In this case, two separate backup procedures have to be done - separately for the database, separately for the file storage.

3. Hybrid storage option. With this storage option, some files can be stored in the database, and some in the file system. When applying such a storage scheme, as a rule, the following approach is used: small files will be stored in the database, large files are stored in the file system. This approach eliminates a number of drawbacks, which are available in the first two options - reducing the time of connection to the database, because the files are small, and when working with large files, the resources of the file system are used, which speeds up the work with these files.

The second important issue in storing binary data is the question of eliminating their duplication. This problem occurs because the same binary file may be required in different places. The optimal solution in this case is the option that the file is placed once, and then references are made to it. However, the implementation of this solution has a number of problems associated with the fact that, firstly, users do not always remember which files they previously downloaded and downloaded the same file several times, and, secondly, users can download files that are were downloaded by other users and did not know about them.

To solve this problem, you must have a mechanism for checking the presence of the downloaded file in the repository. To do this, you need to implement the procedure for verifying the existence of an identical file in the repository. If the file already exists in the repository, the new file is not added to the repository, and the user is returned a pointer to the already existing file. Otherwise, the file is added to the vault and the user returns a pointer to the downloaded file.

When implementing the procedure for verifying the presence of an identical file in the store, a problem arises due to the number and size of the available files in the repository. This is due to the fact that you need to go through the entire

list of files and reconcile the contents of the downloaded file with the existing files in the repository. If you have a large number of files, this process can take a long time.

One of the effective mechanisms for optimizing file search is the use of cryptographic hash functions. A feature of these hash functions is that when they input byte length data of arbitrary length, they return a byte value of a fixed length. The organization of the search system when using cryptographic hash functions can have the following sequence:

1. The user downloads the file to the system
2. The content of the received file is fed to the input of the hash function
3. The received hash value searches the database for available files
4. If the hash value is present in the database, then the pointer to the existing file is returned, otherwise the file is added to the repository, and the hash of the new file is added to the database.

Hash values that return cryptographic hash functions are small. If these values are stored in a table in the database, then this will create an index on the field where this value is stored, which will dramatically increase the speed of searching for files.

The system of distance learning on the educational portal of the D.Serikbayev EKSTU

The EKSTU educational portal has a distance learning system. This system is part of the software package of the educational portal SPortal / Dales. At the software level, this system is implemented as a web-application based on Microsoft ASP.NET technology version 4.0. This web-application is working within the framework of the Microsoft Internet Information Services web-service, which is included in the Microsoft Windows 2012 operating system. As a database for the distance learning system, the educational portal database acts. This database is based on the database management system Microsoft SQL Server 2012.

Access to work with this system is authorized, and the following categories of users can access it:

1. Students
2. Teachers
3. Decanting
4. Training part

The main component of this system is the training course. The training courses are tied to discipline and the academic group that studies this discipline. Access to a specific distance course is available only to learning groups, teachers, leading discipline and administrative staff.

Each distance course has the following hierarchical structure:

– Block - combines several weeks. The blocks are: boundary control #1, boundary control #2, examination block. For each block, the final score is calculated as the arithmetic mean for all estimates obtained from the tasks included in this block
  • A week is a unit that includes various resources and tasks that should be studied by students in the given week.
    * Resource is a kind of educational material necessary for training. The following resources are available for the course:
      · File is a file that contains the information required to study the course
      · Link to external resource is a URL address to an external resource on which there is any information for studying within the course
      · Text block is a textual content for explaining to students at the course
    * Task is a task that must be performed by trainees. For each completed task, the trainee receives an assessment. The task can be ordinary (the evaluation is taken into account when calculating the final score for the block) and training (the estimate is not taken into account when calculating the final block score). The assignment must be carried out within the time frame indicated by the teacher. The following types of tasks are available:
      · Practical / laboratory work is a practical task that the learner must perform. The content of the task is downloaded as a file. The trainee performs this task and as a response must attach the file with the completed task, which the teacher must evaluate.
      · Forum is a discussion of the topic on the course forum. As a result of the discussion, the teacher presents the students with grades.
      · Chat is a discussion of the topic in the course chat. As a result of the discussion, the teacher presents the students with grades.
      · Testing is the execution of a test task with the choice of one answer from several. The task is evaluated automatically according to the results of the students' answers. In case of disagreement with the results, the student may appeal the questions that were incorrectly answered. An appeal is considered by the course instructor and, if satisfied, the assessment of the test will be changed.

Organization of storing binary data in the system of distance learning

As mentioned in the previous section, Microsoft SQL Server 2012 is used as a database management system. This database management system has a special technology for storing files from tables in the file system. This technology is called

Filestream. This technology allows you to work with files as with conventional binary data inside tables, as well as with regular files. This technology is used to organize the storage of binary data in the distance learning system on the educational portal.

To solve the problem of file duplication, which was described in the item "Issues related to storing binary data," we used a solution based on the use of cryptographic hash functions. As a hash function, we used a hash scheme based on the SHA-1 algorithm. This algorithm generates binary data of a fixed size of 20 bytes in the output.

Based on the description of the distance course structure given in the previous section, the binary data that is required to be stored in the distance learning system are:

1. Resource - File;
2. Task - Practical / laboratory work

Figure 1 shows the structure of the part of the database intended for storing the data of the course
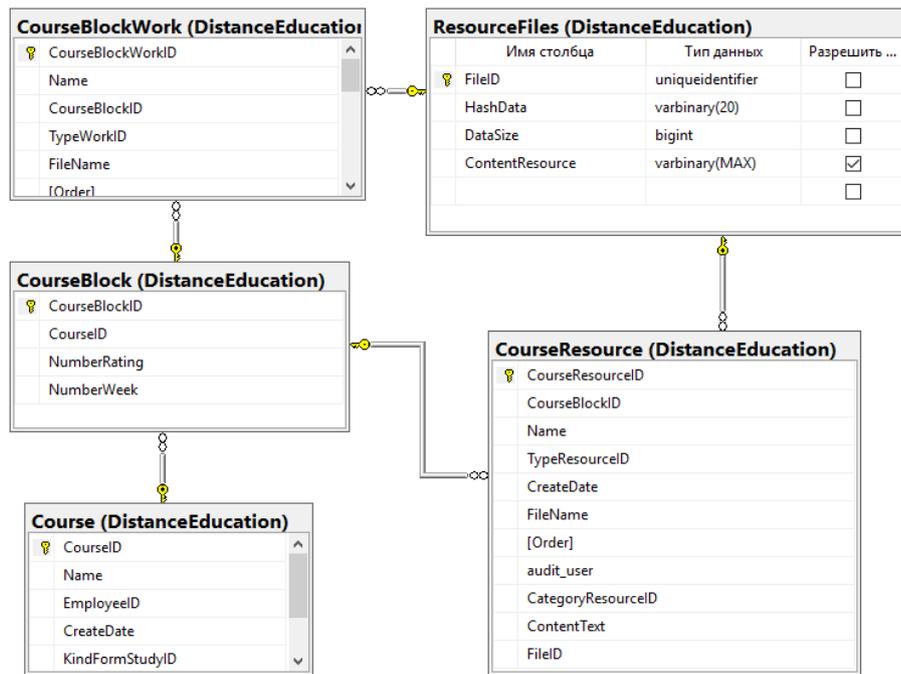
Figure 1. Structure of the table for storing remote distance data

Figure 1 shows the following tables:

1. Course - a table describing the course
2. CourseBlock - a table describing the blocks and weeks
3. CourseResource - a table that describes the resources for the course
4. CourseBlockWork - a table that describes the tasks at the course
5. ResourceFiles - a table representing binary data stored using Filestream technology.

Consider the structure of the ResourceFiles table in more detail. This table has the following fields:

1. FileID, which has a uniqueidentifier type and is the key table. This field of this type is mandatory for the organization of work with the technology of Filestream.
2. HashData, which has a type of binary data of 20 bytes in size. This field is intended for storing the calculated hash value for binary data by the SHA-1 algorithm. A unique index is created for this field, which guarantees duplication of binary data in the table
3. DataSize - a field that specifies the size of stored binary data
4. ContentResource - a field for storing binary data based on the Filestream technology.

Table ResourceFiles, as seen in the diagram in Figure 1, associated with the tables CourseResource and CourseBlockWork on the field FileID. This relationship helps determine which files are tied to resources and practical tasks.

Working with binary data from the web application of the distance learning system

As described above, the Filesteam technology allows you to work with binary data not only as binary data within a database, but also as with regular files. For such work .Net Framework there is a special class SqlFileStream. This class is used by us to organize work with binary data in the web application of the distance learning system. Listing 1 shows a listing from a web application designed to work with binary data in a database.

Listing 1 - working with binary data

```
public void ReadFromFileStream(string filePath,
    Stream writeStream, int bufSize)
{
    SqlTransaction tran = mConnection.BeginTransaction();
    SqlCommand cmd_tran = mConnection.CreateCommand();
    cmd_tran.Transaction = tran;
    cmd_tran.CommandText =
        "SELECT GET_FILESTREAM_TRANSACTION_CONTEXT()";
    byte[] tran_data = (byte[])cmd_tran.ExecuteScalar();
    using (SqlFileStream fs = new SqlFileStream(filePath,
```

```
        tran_data, FileAccess.Read))
    {
        byte[] buffer = new byte[bufSize];
        using (BinaryWriter bw = new BinaryWriter(writeStream))
        {
            int rb = 0;
            do
            {
                rb = fs.Read(buffer, 0, bufSize);
                bw.Write(buffer, 0, rb);
            } while (rb == bufSize);
            bw.Close();
        }
        fs.Close();
    }
    tran.Commit();
}

public void ReadFromFileStream(string filePath,
    Stream writeStream)
{
    ReadFromFileStream(filePath, writeStream, 4096);
}
public Guid WriteResourceFileStream(byte[] file)
{
    Guid FileID;
    SHA1 hashAlg = SHA1.Create();
    byte[] hash = hashAlg.ComputeHash(file);
    object rq = Scalar("SELECT FileID
                        FROM DistanceEducation.ResourceFiles
                        WHERE HashData = @HashData AND
                              DataSize = @DataSize",
        Parameter("@HashData", hash),
        Parameter("@DataSize", file.Length));
    if (rq == null)
    {
        FileID = Guid.NewGuid();
        SqlTransaction tran = mConnection.BeginTransaction();
        SqlCommand cmd = mConnection.CreateCommand();
        cmd.Transaction = tran;
        cmd.CommandText =
            "SELECT GET_FILESTREAM_TRANSACTION_CONTEXT()";
        byte[] tranID = (byte[])cmd.ExecuteScalar();
        SqlCommand cmd_ins = mConnection.CreateCommand();
        cmd_ins.Transaction = tran;
```

```
    cmd_ins.CommandText =
        "INSERT INTO DistanceEducation.ResourceFiles
          (FileID,HashData, DataSize, ContentResource)
          VALUES(@FileID, @HashData, @DataSize, 0x)";
    cmd_ins.Parameters.AddWithValue("@FileID", FileID);
    cmd_ins.Parameters.AddWithValue("@HashData", hash);
    cmd_ins.Parameters.AddWithValue("@DataSize",
        file.Length);
    cmd_ins.ExecuteNonQuery();
    SqlCommand cmd_path = mConnection.CreateCommand();
    cmd_path.Transaction = tran;
    cmd_path.CommandText =
        "SELECT ContentResource.PathName(2)
          FROM DistanceEducation.ResourceFiles
          WHERE FileID = @FileID";
    cmd_path.Parameters.AddWithValue("@FileID", FileID);
    string path = (string)cmd_path.ExecuteScalar();
    SqlFileStream fs = new SqlFileStream(path, tranID,
        FileAccess.Write);
    fs.Write(file, 0, file.Length);
    fs.Close();
    tran.Commit();
}
else
{
    FileID = (Guid)rq;
}
return FileID;
}
```

As you can see from the above listing, before starting to work with SqlFileStream, we must start the transaction and execute the command

GET_FILESTREAM_TRANSACTION_CONTEXT

This command returns a special object that is used to work with SqlFileStream.

In the above listing, we have two functions:

1. ReadFromFileStream - to read data
2. WriteResourceFileStream - to write data. When writing data, as seen from the listing, we first compute the hash, check the presence of similar data in the ResourceFiles table. If there is no such binary data, then we are adding a new file to the database.

Conclusion

In this article, we:

1. Considered the problems that arise when storing binary data;
2. Described approaches that can be used to solve the issues of data storage organization;
3. They described the system of distance learning, which was implemented on the educational portal of the EKSTU
4. We examined how the scheme for storing binary data in a database is organized and how the work with them has been implemented in the web application of the distance learning system

References

1. Bondar A.G., *Microsoft SQL Server 2012*. – St. Petersburg: BHV-Petersburg, 2013. – 608 p.
2. Koneev I.R., Belyaev A.V., *Information security of the enterprise*. – St. Petersburg: BHV-Petersburg, 2003. – 747 p.

Automating class scheduling for the academic portal of the university

S. K. Kumargazhanova, G. Zhomartkyzy, E. M. Fedkin, and
Zh. T. Konurbayeva

D. Serikbayev EKSTU, Oskemen, Republic of Kazakhstan
`skumargazhanova@gmail.com`,`GZhomartkyzy@ektu.kz`,`efedkin@mail.ru`

Abstract This article discusses the problem of class scheduling in universities. This problem belongs to the class of NP-complete problems. To solve this problem, we can use such a heuristic method as a genetic algorithm. This algorithm was implemented on the educational portal at D. Serikbayev EKSTU to compile class schedules.

Keywords: class scheduling, "hard"and "soft"restrictions, genetic algorithm, crossing, mutation, fintess function

Introduction

The preparation of classes is one of the main tasks in the organization of the educational process at the university. This problem concerns the problems of integer programming. The complexity of the task of scheduling increases with the number and range of possible values of variables that are used to compile the schedule. Such problems belong to the class of NP-complete problems, the solution of which requires a lot of time [1].

Considering the fact that the development of precise scheduling algorithms is a rather complex task, for its solution it is most often based on the application of heuristic methods. The application of these methods for scheduling is based on the use of heuristics or heuristic algorithms that use intuitive assumptions that do not have a clear mathematical justification. The use of heuristic methods for scheduling on the basis of heuristic rules makes it possible to speed up the process of finding the schedule closest to optimal, although it does not guarantee obtaining an optimal timetable [2].

Settings of class scheduling

The initial variables in the composition of the schedule are the following sets:

1. $A \in (A_1, A_2, \ldots, A_n)$ is a set of classrooms that can be used for classes. Each audience is characterized by a number of characteristics such as capacity, availability of equipment required to conduct a particular class of activities, etc.

2. $P \in (P_1, P_2, \ldots, P_n)$ is a set of university teachers who will conduct classes according to the schedule.
3. $D \in (D_1, D_2, \ldots, D_n)$ is a set of disciplines that will be included in the schedule.
4. $G \in (G_1, G_2, \ldots, G_n)$ is a set of groups, which will be held classes.
5. $S \in (S_1, S_2, \ldots, D_n)$ is a set of subgroups on which the group can be divided during the conduct of classes. Some types of classes in disciplines, such as laboratory and practical classes, have limitations on the number of students for whom the activity is being conducted. In the event that the number of students exceeds this limit, the group for conducting the lesson is divided into two or more subgroups.
6. $W \in (W_1, W_2, \ldots, W_n)$ is a set of days of the week in which classes are held.
7. $T \in (T_1, T_2, \ldots, T_n)$ is a set of time intervals during the day in which lessons are held.
8. $K \in (K_1, K_2, \ldots, K_n)$ is a set of types of classes that can be conducted in discipline (lectures, practices, laboratory, etc.)
9. $F \in (F_1, F_2, \ldots, F_n)$ is the set of academic flows. Each academic stream will combine the discipline, the teacher, the group, the subgroup and the type of occupation - $F = (D, P, G, S, K)$.

Based on the sets listed above, the goal of class scheduling is to find the next set

$$R = (F, A, W, T) \tag{1}$$

Elements of a given set must satisfy a number of constraints. These restrictions are divided into 2 large groups [3]:

1. "Hard" restrictions are requirements that must necessarily be met. Such requirements include:

   - Lack of overlays for the audience, i.e. in one classroom there can not be two or more different classes;
   - no overlap for the group, i.e. group / subgroup can not be in different audiences at the same time
   - lack of overlays for the teacher, i.e. The teacher can not simultaneously conduct classes in different classrooms
   - accounting for the capacity of the audience, i.e. At the same time, there can not be more students in the classroom than the audience capacity
   - availability of equipment in the classroom, i.e. The classroom should have the necessary equipment required for conducting the class
   - other mandatory restrictions.

2. "Soft" restrictions are desires of teachers, students and other interested persons to the schedule of classes. These requirements are recommendatory in nature and their implementation is not mandatory.

Genetic algorithm for scheduling

One of the most effective heuristic algorithms used in scheduling is the genetic algorithm. The given algorithm has a number of advantages at drawing up of the schedule before other heuristic algorithms [2]:

1. the algorithm works with codes representing a set of parameters that are arguments of the objective function;
2. the algorithm operates with the whole set of possible solutions of the problem;
3. the operation of the algorithm does not require additional information, which greatly speeds up the search speed of the solution;
4. the algorithm uses probabilistic and deterministic rules to generate new search points.

The basic unit with which the genetic algorithm works is the chromosome. The chromosome is a possible solution to the problem. The chromosome consists of genes that represent the parameters of the problem. Based on our formulation of the task of scheduling, the chromosome will represent the value of R_i - a possible schedule of employment, and the genes will be the constituents of R_i. The number of genes is determined by the number of sessions for which a schedule is needed.

The search for a solution in the genetic algorithm goes through several stages [3].

1. Formation of the initial population. At this stage, a necessary number of individuals with a given set of chromosomes is randomly created.
2. Crossing. At this stage, two individuals are randomly selected. Further, these two individuals randomly exchange the corresponding genes.
3. Mutation. At this stage, the values in the genes are randomly changed to other acceptable values.
4. Selection of individuals. At this step, we select the individuals most fit for the next steps in the algorithm. A special function is used for selection, which is called fitness function. In our case, this function is defined as the implementation of chromosomes of "hard"and "soft"constraints based on weighting coefficients. The adaptability function in this case will have the following form:

$$f(a) = \sum_{i=1}^{n} k_i h_i + \sum_{j=1}^{m} k_j h_j \qquad (2)$$

where, a - the individual for which the value of the function is calculated,

h_i − execution of the "hard"constraint for the schedule (takes two values: 0 - the condition is not fulfilled, 1 - the condition is fulfilled),

s_j − execution of a "soft"constraint for the schedule (takes two values: 0 - the condition is not fulfilled, 1 - the condition is fulfilled),,

k_i − weighting factor for constraint.

5. Formation of a new generation. As a result of applying the actions at the stages of breeding, crossing and mutation, a new generation of individuals is formed.

6. Checking the conditions for stopping the algorithm. For individuals of the new generation, the condition for completing the search is checked. In our case, we use the following condition to stop the search:

   - all "tight"constraints are met
   - the specified percentage of individuals has the value of the adaptability function above the established one.

   If the search termination condition is not met, then the second stage of the algorithm operation is performed, otherwise the transition to the final stage is performed.

7. Choosing the "best"individual. At this stage, the "best"individual is selected in the last generation based on the value of the adaptability function. This individual will be the solution of the search problem, i.e. in our task - the desired optimal schedule of classes.

The graphical genetic algorithm for scheduling occupations is shown in Figure 1.
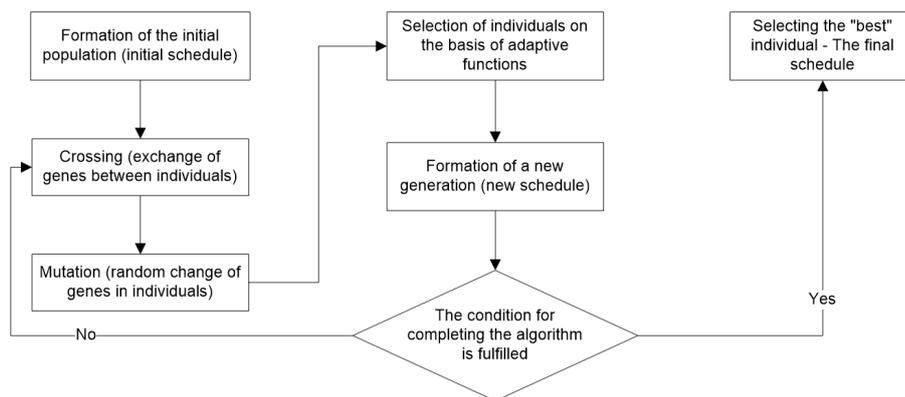
Figure 1. Scheme of the genetic algorithm

Modification of the algorithm to compile a partial schedule

It is often required to draw up a schedule of classes not completely, but in part. An example of such a timetable is the timetable, which is compiled for some disciplines not for the entire semester, but only for a part of the semester. In this case, we can use the following scheme for scheduling classes:

1. First, schedules of classes on disciplines are formed, which will be conducted throughout the whole semester
2. Forming a schedule of classes that are part of the semester. In this case, classes that are conducted for the entire semester or partially overlap with the period being formed with the schedule are included in the genetic algorithm as unchangeable genes. Then the algorithm should work according to the standard scheme.

Realization of scheduling of lessons on the educational portal of D. Serikbayev EKSTU

There is a module for scheduling classes on the academic portal of D. Serikbayev EKSTU. With the help of this module, the dispatch service has the possibility of scheduling classes by groups and teachers. The schedule editor is implemented in the Delphi environment and is part of the SPortal software package. Based on the completed schedule on a schedule in the context of groups, teachers, departments and faculties. Also, on the basis of the schedule, reporting forms are generated according to the occupancy of the audiences. These reporting forms are implemented on the basis of dynamic html-pages in the system "Dales: The Methodists which operates on the basis of the web server Microsoft Internet Information Services.

The genetic algorithm for scheduling lessons described in the previous section is implemented in the SPortal software package. It allows you to generate a new training schedule, which the dispatch service can then manually adjust in the schedule editor (Figure 2).

Conclusion

In this article we:

1. Describe the approaches to scheduling classes in universities;
2. The description of setting the task of scheduling was given;
3. Proposed a genetic algorithm for scheduling lessons based on the task. For the algorithm, the adaptability function was defined, the condition for completing the search and selecting the "best"individual based on the weighting coefficients of the constraints for the schedule.
4. The described algorithm was implemented in the software package SPortal to compile an initial schedule of classes on the educational portal of D. Serikbayev EKSTU.

References

1. Sidorin A.B., Lykucheva L.V., Dvoryankin A.M., *Automation methods of scheduling exercises Part 1. Classical methods*, T.7, #12 (60), Izvesti of Volgograd State Technical University, 116-120 (2009)).
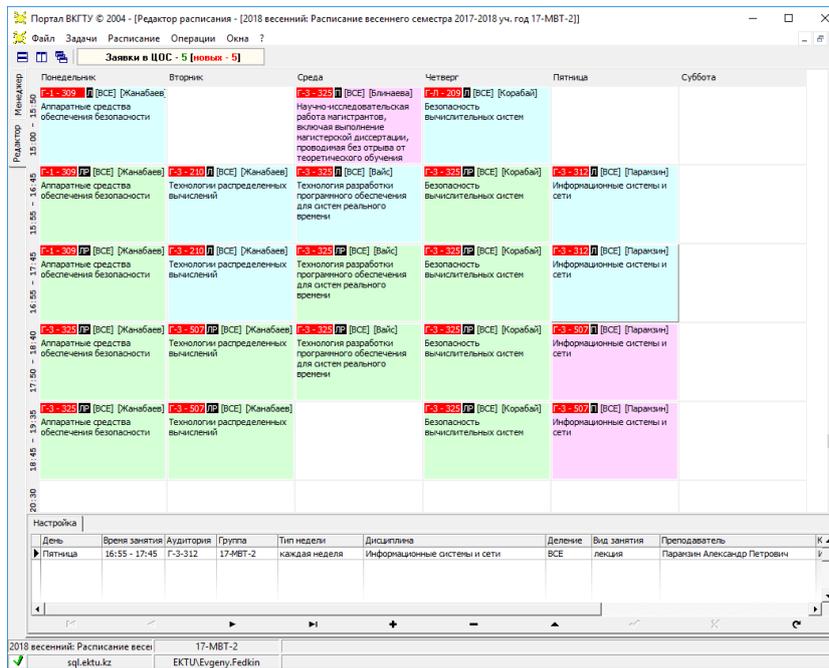
Figure 2. Schedule editor

2. Sidorin A.B., Lykucheva L.V., Dvoryankin A.M., *Automation methods of scheduling exercises Part 2. Heuristic optimization methods*, T.7, #12 (60), Izvesti of Volgograd State Technical University, 120-123 (2009).
3. Kabalnov Yu.S., Shekhtman L.I., Nizamova G.F., Zemchenkova N.A., *Compositional genetic algorithm for scheduling training sessions*, Bulletin of Ufa State Aviation Technical University, Vol. 7, #2, 99-109 (2006).

Numerical simulation of convective flows of a viscous incompressible fluid in curvilinear multiply connected domains

E. A. Malgazhdarov[1], N. M. Temirbekov[2], and S. O. Tokanova[1]

[1] D. Serikbayev East Kazakhstan State Technical University,
19 Serikbayev, Ust-Kamenogorsk, Kazakhstan
[2] Kazakhstan Engineering Technological University
93G al-Farabi, Almaty, Kazakhstan
telec_saya@mail.ru

Abstract In this paper we describe a method for the numerical construction of curvilinear structured grids in doubly connected domains and numerical modeling of the convective flow of a nonuniformly heated liquid in a curvilinear coordinate system. The implicit scheme and the method of fractional steps are used in the numerical construction of curvilinear grids in doubly connected domains by equidistribution methods and the method of Godunov-Thompson. An explicit scheme and the method of fractional steps are used in the numerical implementation of the equations of an incompressible fluid. A cyclic run is used in the direction of the outer and inner boundaries, and a scalar run is used in the normal direction. Calculations were carried out for different cavity configurations and temperature regimes at the boundary. The plots of numerical calculations of the temperature and current function are obtained.

Keywords: computer technology, mathematical modeling, curvilinear structured grids, doubly-connected, curvilinear boundary

Introduction

With the rapid development of computer technology, mathematical modeling of physical, chemical processes and mechanical systems in various branches of science is intensively developing. In recent years, it has become increasingly necessary to solve problems in complex regions with complex geometry and in zones of rapid changes in the characteristics of the physical medium (density, pressure). For modeling in complex areas, first of all, the physical area is to be discretized, that is, the stage of modeling physical geometries using a set of cells of difference grids. It should be noted that the use of non-uniform grids can cause non-physical sources of mass and momentum to appear in the calculation schemes, as well as the loss of important properties inherent in the approximated differential equations. The model equations recorded in curvilinear coordinates have a more complex form than the original equations, in particular,

they contain variable coefficients, additional terms, nonzero right-hand sides, and so on. Therefore, the question of approximation of equations on curvilinear grids is urgent and requires close attention. In addition, the diverse requirements imposed on difference grids make the construction of curvilinear grids a complex mathematical problem. In this regard, the development of theoretical concepts and methodological approaches to the use of advanced information technologies in hydrodynamic studies that take into account the specific features of the subject area, the development, adaptation of tools and approbation of them in the process of modeling of natural and technogenic objects of significant economic importance are very relevant.

Modern research in the field of computational and applied mathematics is aimed at the creation of automated computer programs for the construction of curvilinear adaptive structural and nonstructural difference grids, as well as the modernization of numerical algorithms for solving applied problems. A fundamental study on the justification, numerical realization of the construction of curvilinear adaptive grids, the problem of hydrodynamics, and also some experimental results were published in the works of N.T. Danaeva, Yu.I. Shokin, G.S. Khakimzyanova, N.M. Temirbekova, V.D. Liseikina, J.F. Thompson, Z.U.A. Warsi, C.W. Mastin, etc. In this paper we consider the problem of constructing curvilinear grids on an arbitrary curvilinear boundary and inside a domain by a differential method. In these methods, differential equations of partial derivatives of various types are used, but differential methods for constructing grids based on solving equations of elliptic type are most widely used. A mathematical problem is also considered with respect to the vorticity variables ω, the stream function ψ and the temperature θ describing the convective flow of a nonuniformly heated viscous fluid in an arbitrary doubly-connected domain with a curvilinear boundary $\partial D = \Gamma_0 \cup \Gamma_1$ (Fig.1).
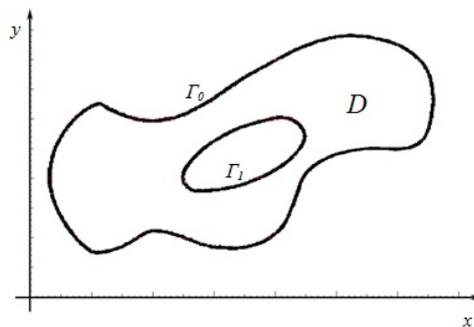
Figure 1. A doubly connected physical domain

Formulation of the Problem

Methods for constructing curvilinear grids are considered. Differential methods for constructing curvilinear grids are used in this paper, since the physical domain under consideration is complex and has curvilinear boundaries. In these methods, partial differential equations of various types are used, but differential methods for constructing grids based on solving equations of elliptic type are most widely used. The mapping of the physical region in the coordinate system (x, y) to the computational domain in the coordinate system (ξ, η) is performed by the method of cutting the region [1] (Fig. 2). The curve of the outer boundary 1 is mapped onto the line $\eta = 0, 0 \leq \xi \leq 1$, and the inner boundary 3 is mapped onto the line $\eta = 1, 0 \leq \xi \leq 1$. The cut along the cutting line is made twice, the boundary 2 is mapped onto the line $\xi = 1, 0 \leq \eta \leq 1$, and the cut line 4 is mapped onto the line $\xi = 0, 0 \leq \eta \leq 1$.
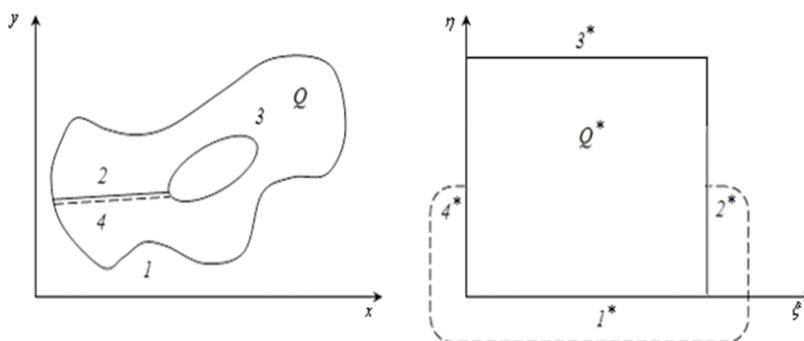
Figure 2. The mapping of a doubly-connected curvilinear domain Q to a calculated rectangle Q *

Computational Algorithm

The construction of a grid in a two-dimensional domain begins with the construction of a grid on its boundary. Since the boundary of the domain is not monotonous, we describe the boundary with the use of equations in a given parametric form:

$$x = f^1(p), y = f^2(p), 0 \leq p \leq l, \tag{1}$$

where l is the length of the boundary. The one-dimensional equidistribution method is used to construct a grid on the boundaries, i.e. a differential equation of the following form [1]:

$$\frac{\partial}{\partial \xi}(\vartheta(p)\frac{\partial p}{\partial \xi}) = 0, \xi \in (0,1) \qquad (2)$$

p(0)=0, p(1)=l, where $\vartheta(p) = \sqrt{\left(\frac{\partial f^1(p)}{\partial p}\right)^2 + \left(\frac{\partial f^2(p)}{\partial p}\right)^2} > 0, p \in [0,l]$

The coordinates of the grid nodes at the boundaries are calculated by formula (1) using the values of found p. The equations of the equidistribution method are used to construct two-dimensional grids with the assumption of orthogonality of the required coordinate system [1]:

$$\frac{\partial}{\partial \xi}\left(g_{22}\frac{\partial \overrightarrow{x}}{\partial \xi}\right) + \frac{\partial}{\partial \eta}\left(g_{11}\frac{\partial \overrightarrow{x}}{\partial \eta}\right) = 0 \qquad (3)$$

where $\overrightarrow{x} = (x,y)$ is physical coordinates, $g_{11} = x_\xi^2 + y_\xi^2, g_{22} = x_\eta^2 + y_\eta^2$ are the components of the metric tensor.

The problem (1)-(3) for the construction of a grid on the boundary of the domain is solved by a finite difference method. The finite difference scheme (2) has the following form:

$$\frac{1}{h_1}\left(\vartheta_{i+\frac{1}{2}}\frac{p_{i+1} - p_i}{h_1} + \vartheta_{i-\frac{1}{2}}\frac{p_i - p_{i-1}}{h_1}\right) = 0; p_1 = 0, p_{n_1} = l; j = 2, ..., n_1 - 1 \quad (4)$$

where $\vartheta_{i+\frac{1}{2}} = \sqrt{\left(\frac{f^1(p_{i+1}) - f^1(p_i)}{p_{i+1} - p_i}\right)^2 + \left(\frac{f^2(p_{i+1}) - f^2(p_i)}{p_{i+1} - p_i}\right)^2}$

If the boundary of the region is given as a set of points $A_k(x_k, y_k), (k = 1, ..., M), (x_k, y_k) \in \Gamma_l(l = 1,2)$, then the length is defined as follows:

$$l_1 = 0; l_k = \sum_{i=2}^{k} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}; k = 2, ..., M.$$

The parametric equation for determining the nodes coordinates on the boundary for linear interpolation and $(p_i \in [l_k, lk+1]$ has the following form:

$$f^1(p_i) = x_k + \frac{x_{k+1} - x_k}{l_{k+1} - l_k}(p_i - l_k), f^2(p_i) = y_k + \frac{y_{k+1} - y_k}{l_{k+1} - l_k}(p_i - l_k). \quad (5)$$

The resulting finite difference problem (4) is solved by an iterative method of successive approximations. A uniform grid on the interval $[0,l]$ is chosen as an initial approximation p_i^0. Let the grid p_i^n be built on the n-th iteration. Let us define

$$\vartheta_{n+\frac{1}{2}} = \sqrt{\left(\frac{f^1(p_{i+1}^n) - f^1(p_i^n)}{p_{i+1}^n - p_i^n}\right)^2 + \left(\frac{f^2(p_{i+1}^n) - f^2(p_i^n)}{p_{i+1}^n - p_i^n}\right)^2}$$

on this grid. The following successive approximation is found using them. For this, the following linear problem is solved

$$\frac{1}{h_1}\left(\vartheta_{i+\frac{1}{2}}^n \frac{p_{i+1}^{n+1} - p_i^{n+1}}{h_1} + \vartheta_{i-\frac{1}{2}}^n \frac{p_i^{n+1} - p_{i-1}^{n+1}}{h_1}\right) = 0; \tag{6}$$

where $p_n^{n+1} = 0, p_{n_1}^{n+1} = l; i = 2, ..., n_1 - 1$.

The iterative process continues to the specified accuracy, that is, until the following condition is fulfilled:

$$\max_{l \le i \le n_1} |p_i^{n+1} - p_i^n| \le \varepsilon.$$

The coordinates of the nodes on the boundary of the physical region are calculated using (5) based on the results of the last iteration approximation. Figures 3 show the results of the solution to the difference problem (6) and (5), where 20, 50 and 100 grid nodes are uniformly distributed at the boundaries.

a) Uniformly distributed mesh nodes (20 nodes)   б) Uniformly distributed mesh nodes (50 nodes)

в) Uniformly distributed mesh nodes (100 nodes)
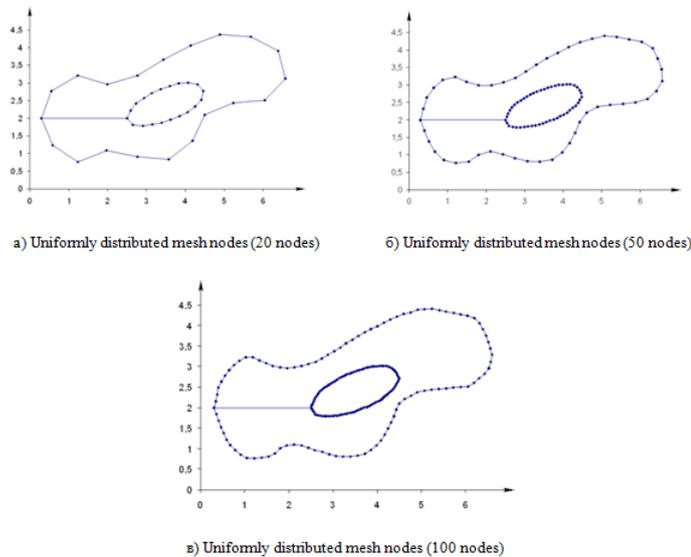
Figure 3. Uniformly distributed mesh nodes

Let us write out the difference problem for determining the nodes coordinates inside the domain. The finite difference scheme (3) has the following form:

$$\Lambda_{11} \vec{x}_{i,j} + \Lambda_{22} \vec{x}_{i,j} = 0; \tag{7}$$

where

$$\Lambda_{11}\, \vec{x}_{\,i,j} = \frac{1}{h_1}\left(g_{22,i+\frac{1}{2},j}\frac{\vec{x}_{\,i+1,j} - \vec{x}_{\,i,j}}{h_1} - g_{22,i-\frac{1}{2},j}\frac{\vec{x}_{\,i,j} - \vec{x}_{\,i-1,j}}{h_1}\right),$$

$$\Lambda_{22}\, \vec{x}_{\,i,j} = \frac{1}{h_2}\left(g_{11,i,j+\frac{1}{2}}\frac{\vec{x}_{\,i,j+1} - \vec{x}_{\,i,j}}{h_2} - g_{11,i,j-\frac{1}{2}}\frac{\vec{x}_{\,i,j} - \vec{x}_{\,i,j-1}}{h_2}\right).$$

To determine the components of the metric tensor, the central differences in the integer nodes

$$x_{\xi,i,j} = \frac{x_{i+1,j} - x_{i-1,j}}{2h_1}, x_{\eta,i,j} = \frac{x_{i,j+1} - x_{i,j-1}}{2h_2}$$

$$y_{\xi,i,j} = \frac{y_{i+1,j} - y_{i-1,j}}{2h_1}, y_{\eta,i,j} = \frac{y_{i,j+1} - y_{i,j-1}}{2h_2}$$

$$g_{11,i,j} = x_{\xi,i,j}^2 + y_{\xi,i,j}^2, g_{22,i,j} = x_{\eta,i,j}^2 + y_{\eta,i,j}^2,$$

and in the center of the faces the cells are determined by averaging in the following way:

$$g_{11,i+\frac{1}{2},j} = \frac{g_{11,i+1,j} + g_{11,i,j}}{2}, g_{11,i-\frac{1}{2},j} = \frac{g_{11,i,j} + g_{11,i-1,j}}{2}.$$

The remaining coefficients are determined similarly. The alternating directions method is used to find the numerical solution of (6). Let us consider the algorithm of the alternating directions method:

$$\frac{\vec{x}_{\,i,j}^{\,n+\frac{1}{2}} - \vec{x}_{\,i,j}^{\,n}}{0.5\tau} = \Lambda_{11}^n\, \vec{x}_{\,i,j}^{\,n+\frac{1}{2}} + \Lambda_{22}^n\, \vec{x}_{\,i,j}^{\,n}; \tag{8}$$

$$\frac{\vec{x}_{\,i,j}^{\,n+1} - \vec{x}_{\,i,j}^{\,n+\frac{1}{2}}}{0.5\tau} = \Lambda_{11}^n\, \vec{x}_{\,i,j}^{\,n+\frac{1}{2}} + \Lambda_{22}^n\, \vec{x}_{\,i,j}^{\,n+1}. \tag{9}$$

Here, n is the number of the iteration, τ is the iteration parameter. Since the components of the metric tensor depend on the solution, the coefficients $g_1 1, g_2 2$ are calculated using the n-th iteration solution. Since the domain under consideration is doubly connected and the grid nodes must coincide on the cut line (see Fig. 2), then in the ξ direction we need to apply the cyclic sweep method [6] with periodic conditions:

$$A_{i+n_1} = A_i, B_{i+n_1} = B_i, C_{i+n_1} = C_i, F_{i+n_1} = F_i, i = 1, ..., n_1 - 1. \tag{10}$$

If conditions (10) are satisfied, the solution of equations (8) is also periodic with period $n_1 - 1$, i.e.

$$\vec{x}_{\,i} = \vec{x}_{\,i,n_1-1}$$

Therefore, it is sufficient to find a solution $\overrightarrow{x}_i, i = \overline{1, n_1 - 1}$. In this case, equation (8) with periodic conditions can be written as follows:

$$
\begin{cases}
-A_1 \overrightarrow{x}_{n_1-1,j}^{n+\frac{1}{2}} + C_1 \overrightarrow{x}_{1,j}^{n+\frac{1}{2}} - B_1 \overrightarrow{x}_{2,j}^{n+\frac{1}{2}} = F_1, \ i = 1 \\
-A_i \overrightarrow{x}_{i-1,j}^{n+\frac{1}{2}} + C_i \overrightarrow{x}_{i,j}^{n+\frac{1}{2}} - B_i \overrightarrow{x}_{i+1,j}^{n+\frac{1}{2}} = F_i, \ 2 \le i \le n_1 - 1 \\
\overrightarrow{x}_{n_1,j}^{n+\frac{1}{2}} = \overrightarrow{x}_{1,j}^{n+\frac{1}{2}}
\end{cases}
\tag{11}
$$

where

$$
A_i = \frac{\tau}{2} \frac{g_{22,i-\frac{1}{2},j}^n}{h_1^2}, B_i = \frac{\tau}{2} \frac{g_{22,i+\frac{1}{2},j}^n}{h_1^2}, C_i = 1 + A_i + B_i, F_i = \overrightarrow{x}_{i,j}^n + \frac{\tau}{2} \Lambda_{22} \overrightarrow{x}_{i,j}^n.
$$

To determine the running coefficients, the following formulas are used [6]:

$$
\alpha_{i+1} = \frac{B_i}{C_i - \alpha_i A_i}; \beta_{i+1} = \frac{F_i + A_i \beta_i}{C_i - \alpha_i A_i}; \gamma_{i+1} = \frac{A_i \gamma_i}{C_i - \alpha_i A_i}; i = 2, ..., n_1 - 1
$$

$$
\alpha_2 = \frac{B_1}{C_1}; \beta_2 = \frac{F_1}{C_1}; \gamma_2 = \frac{A_1}{C_1};
$$

$$
p_i = \alpha_{i+1} p_{i+1} + \beta_{i+1}; q_i = \alpha_{i+1} q_{i+1} + \gamma_{i+1}; i = n_1 - 2, ..., 1;
$$

$$
p_{n_1-1} = \beta_{n_1}; q_{n_1-1} = \alpha_{n_1} + \gamma_{n_1};
$$

$$
\overrightarrow{x}_{n_1,j}^{n+\frac{1}{2}} = \frac{\beta_{n_1+1} + \alpha_{n_1+1} p_1}{1 - \alpha_{n_1+1} q_1 - \gamma_{n_1+1}}; \overrightarrow{x}_{i,j}^{n+\frac{1}{2}} = p_i + q_i \overrightarrow{x}_{n_1,j}^{n+\frac{1}{2}}; i = 2, ..., n_1 - 1.
$$

On the cut line, the following periodic boundary conditions are taken into account:

$$
A_{n_1} = A_1, B_{n_1} = B_1, C_{n_1} = C_1, F_{n_1} = F_1.
$$

A scalar sweep with fixed boundary values found with the help of (5) is used for equation (9) in the direction η. Methodical calculations of the construction of curvilinear grids are considered based on the method described above.

Since the conditions of periodicity were used in one direction, the results are also periodic. In order to determine the most optimal grid, we used the estimation of the grids quality according to the methods proposed in [1]. The work [1] considers four types of estimates that are orthogonality, local uniformity, non-extension and convexity of the constructed grid.

Each grid cell is considered and is divided diagonally into triangles. The following value corresponds to the convexity criterion:

$$
Q_{i,j}^1 = \frac{min \left\{ \begin{matrix} S_{(i,j),(i+1,j),(i+1,j+1)}, S_{(i,j),(i,j+1),(i+1,j+1)}, \\ S_{(i,j),(i+1,j),(i,j+1)}, S_{(i+1,j),(i,j+1),(i+1,j+1)} \end{matrix} \right\}}{0.5(S_{(i,j),(i+1,j),(i+1,j+1)} + S_{(i,j),(i,j+1),(i+1,j+1)})}
\tag{12}
$$

where

$$
S_{(i,j),(i+1,j),(i+1,j+1)} = \frac{1}{2} \left[\begin{matrix} (x_{i+1,j} - x_{i,j})(y_{i+1,j+1} - y_{i,j}) - \\ -(x_{i+1,j+1} - x_{i,j})(y_{i+1,j} - y_{i,j}) \end{matrix} \right],
$$
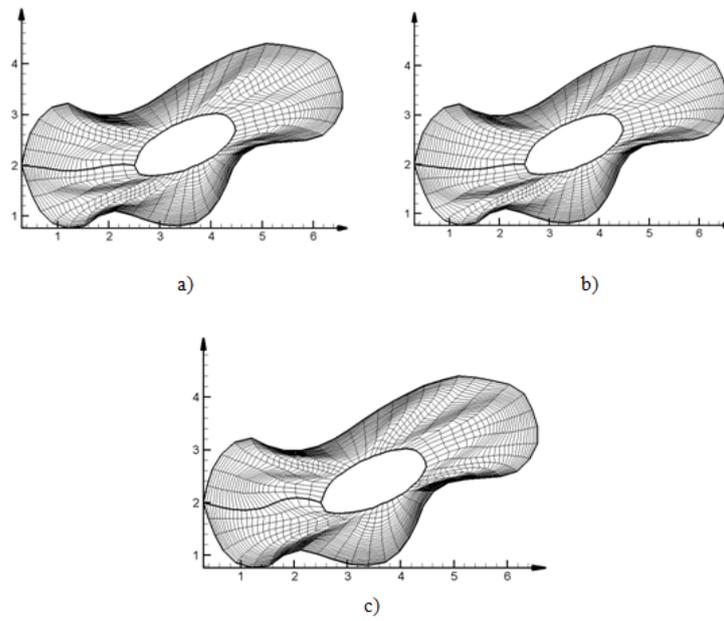
Figure 4. Results of calculations for the construction of curvilinear grids
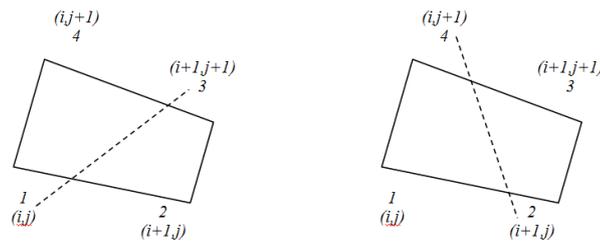
Figure 5. Cell division into triangles

$$S_{(i,j),(i,j+1),(i+1,j+1)} = \frac{1}{2} \begin{bmatrix} (x_{i+1,j+1} - x_{i,j})(y_{i,j+1} - y_{i,j}) - \\ -(x_{i,j+1} - x_{i,j})(y_{i+1,j+1} - y_{i,j}) \end{bmatrix},$$

$$S_{(i,j),(i+1,j),(i+1,j)} = \frac{1}{2} \begin{bmatrix} (x_{i+1,j} - x_{i,j})(y_{i,j+1} - y_{i,j}) - \\ -(x_{i,j+1} - x_{i,j})(y_{i+1,j} - y_{i,j}) \end{bmatrix},$$

$$S_{(i+1,j),(i,j+1),(i+1,j+1)} = \frac{1}{2} \begin{bmatrix} (x_{i+1,j+1} - x_{i+1,j})(y_{i,j+1} - y_{i+1,j}) - \\ -(x_{i,j+1} - x_{i+1,j})(y_{i+1,j+1} - y_{i+1,j}) \end{bmatrix}$$

the areas of the corresponding triangles formed by the diagonals. The value $Q_{i,j}^1$ can lie in the interval $(-\infty, 1]$, for a convex cell $0 < Q_{i,j}^1 \leq 1$, for degenerate in a triangle and self-intersecting cells $-\infty < Q_{i,j}^1 \leq 0$. To determine the estimate of the orthogonality criterion, use the minimum value of the sine of the angle as follows:

$$Q_{i,j}^2 = \min_{k=(i,j),(i+1,j),(i,j+1),(i+1,j+1)} \{\sin \varphi_k\}, \qquad (13)$$

where

$$\sin \varphi_{i,j} = \frac{2S_{(i,j),(i+1,j),(i,j+1)}}{l_{(i,j),(i+1,j)}l_{(i,j)(i,j+1)}}, \sin \varphi_{i+1,j} = \frac{2S_{(i,j),(i+1,j),(i+1,j+1)}}{l_{(i,j),(i+1,j)}l_{(i+1,j)(i+1,j+1)}},$$

$$\sin \varphi_{i,j+1} = \frac{2S_{(i,j),(i,j+1),(i+1,j+1)}}{l_{(i,j),(i,j+1)}l_{(i,j+1)(i+1,j+1)}},$$

$$\sin \varphi_{i+1,j+1} = \frac{2S_{(i+1,j),(i,j+1),(i+1,j+1)}}{l_{(i+1,j),(i+1,j+1)}l_{(i,j+1)(i+1,j+1)}},$$

and the lengths of the sides are $l_{(i,j)(i+1,j)} = \sqrt{(x_{i+1,j} - x_{i,j})^2 + (y_{i+1,j} - y_{i,j})^2}$, and so on.

Values of functions $Q_{i,j}^2$ can take values from a segment $[-1, 1]$; for convex cells it takes positive values, for degenerate ones it vanishes, and for a nonconvex and self-intersecting cell is takes negative values. The next criterion of grid quality is the elongation of the cell, which is defined as follows:

$$Q_{i,j}^3 =$$

$$= \frac{\min_{k=[(i,j),(i+1,j)],[(i+1,j),(i+1,j+1)],[(i+1,j+1),(i,j+1)],[(i,j+1),(i,j)]} \{l_k\}}{\max_{k=[(i,j),(i+1,j)],[(i+1,j),(i+1,j+1)],[(i+1,j+1),(i,j+1)],[(i,j+1),(i,j)]} \{l_k\}}. \qquad (14)$$

Values of $Q_{i,j}^3$ vary in the interval $[0, 1]$. One of the main requirements is local uniformity, i.e. all cells in the area should be uniformly distributed. Adaptive grids are considered in [1], therefore they use a control function. The criterion of local uniformity without the control function is defined as follows:

$$Q_{i,j}^4 = min \left\{ \frac{S_{i+1/2,j+1/2}}{\tilde{S}}, \frac{\tilde{S}}{S_{i+1/2,j+1/2}} \right\}, \qquad (15)$$

where $S_{i+1/2,j+1/2}$ is the area of the cell surrounded by nodes

$$(i,j), (i+1,j), (i+1,j+1), (i,j+1)$$

and is the average area of one cell. The values of

$$\tilde{S} = \frac{\sum_{i=1}^{n_1-1} \sum_{j=1}^{n_2-1} S_{i+1/2,j+1/2}}{(n_1-1)(n_2-1)}$$

vary in the interval $[0.1]$. In order to determine the best variant of the grid, the criteria for the quality of the grid were determined at each iteration according to the methods described above. Based on the defined criteria for the quality of the grid at each iteration, the worst (minimum estimate) was determined, and the best ones were chosen from the worst ones. Thus, the most optimal grid was defined by convexity, since convexity and orthogonality are related criteria.
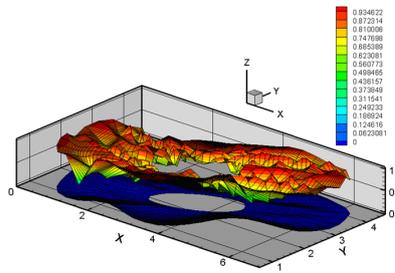
Figure 6. Graphical representation of the convexity criterion in space
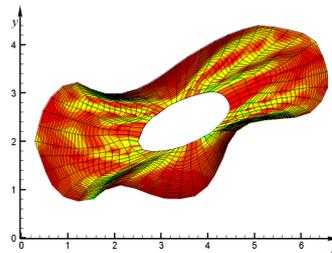
Figure 7. Graphical representation of the convexity criterion in the plane

It can be seen from Figures 8 and 9 that all grid cells are convex, since the values of the estimates are in intervals $0 < Q_{i,j}^1 \le 1$. Thus, we have the

most suitable and mutually orthogonal curvilinear grid in a doubly-connected domain. To simulate a convective flow, the equation of an incompressible fluid is used in the vorticity ω, stream function ψ, and temperature θ variables with corresponding initial and boundary conditions [5] in curvilinear coordinate systems. A mathematical problem describing the convective flow of a nonuniformly heated viscous fluid in an arbitrary doubly-connected domain D with a curvilinear boundary $\partial D = \Gamma_0 \cup \Gamma_1$ is considered (Fig. 1). This problem in Cartesian coordinate system, in a fixed bounded two-dimensional domain can be formulated as follows:

$$\frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + \nu \frac{\partial \omega}{\partial y} = \mu_u \frac{\partial}{\partial x}\left(\frac{\partial \omega}{\partial x}\right) + \mu_u \frac{\partial}{\partial y}\left(\frac{\partial \omega}{\partial y}\right) + \beta \frac{\partial \theta}{\partial x}, \tag{16}$$

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\omega, \tag{17}$$

$$\frac{\partial \theta}{\partial t} + u \frac{\partial \theta}{\partial x} + \nu \frac{\partial \theta}{\partial y} = \mu_\theta \frac{\partial}{\partial x}\left(\frac{\partial \theta}{\partial x}\right) + \mu_\theta \frac{\partial}{\partial y}\left(\frac{\partial \theta}{\partial y}\right), \tag{18}$$

with the following initial and boundary conditions:

$$\omega = 0, \theta = \varphi(x,y), \ (x,y) \in \overline{D}, t = 0,$$
$$\psi = 0, \frac{\partial \psi}{\partial \overline{n}} = 0, \theta = \phi_1(x,y,t), (x,y) \in \Gamma_0, t \in (0,T], \tag{19}$$
$$\psi = \lambda(t), \frac{\partial \psi}{\partial \overline{n}} = 0, \theta = \phi_2(x,y,t), (x,y) \in \Gamma_1, t \in (0,T],$$

where x, y are the Cartesian coordinates, t is time, $u = \frac{\partial \psi}{\partial y}, \nu = -\frac{\partial \psi}{\partial x}$ are the components of the velocity vector, μ_u, μ_θ are the coefficients of kinematic viscosity and thermal diffusivity, β is the coefficient of thermal density variation, Γ_0, Γ_1 are the disjoint contours, $\overline{x}$ is the direction of the outer normal to the boundary $\partial D, \phi i$ is the specified functions. An analogous problem was considered in [4] on a closed rectangular cavity D with a boundary Γ_0 containing a rectangular body with a boundary Γ_1 on a uniform rectangular grid taking into account the uniqueness of the pressure condition. The use of a Cartesian coordinate system in finite difference methods is complicated in a doubly-connected domain with a curvilinear boundary, as in Fig. 1, because of the description of the computational grid at the boundaries and the necessary interpolation procedures for obtaining boundary conditions. The problem (16)-(19) in the curvilinear coordinate system after mapping the physical plane (x,y) to the computational plane (ξ, η) is as follows:

$$\frac{\partial \omega}{\partial t} + a_1 \frac{\partial \omega}{\partial \xi} + a_2 \frac{\partial \omega}{\partial \eta} = J\mu_u \frac{\partial}{\partial \xi}\left(a_{11} \frac{\partial \omega}{\partial \xi}\right) + J\mu_u \frac{\partial}{\partial \eta}\left(a_{22} \frac{\partial \omega}{\partial \eta}\right) -$$
$$- J\mu_u \frac{\partial}{\partial \xi}\left(a_{12} \frac{\partial \omega}{\partial \eta}\right) - J\mu_u \frac{\partial}{\partial \eta}\left(a_{12} \frac{\partial \omega}{\partial \xi}\right) + J\beta\left[y_\eta \frac{\partial \theta}{\partial \xi} - y_\xi \frac{\partial \theta}{\partial \eta}\right] \tag{20}$$

$$\frac{\partial}{\partial \xi}\left(a_{11} \frac{\partial \psi}{\partial \xi}\right) - \frac{\partial}{\partial \xi}\left(a_{12} \frac{\partial \psi}{\partial \eta}\right) -$$
$$\frac{\partial}{\partial \eta}\left(a_{12} \frac{\partial \psi}{\partial \xi}\right) + \frac{\partial}{\partial \eta}\left(a_{22} \frac{\partial \psi}{\partial \eta}\right) = -\frac{\omega}{J} \tag{21}$$

$$\frac{\partial \theta}{\partial t} + a_1 \frac{\partial \theta}{\partial \xi} + a_2 \frac{\partial \theta}{\partial \eta} = J\mu_\theta \frac{\partial}{\partial \xi}\left(a_{11}\frac{\partial \theta}{\partial \xi}\right) + J\mu_\theta \frac{\partial}{\partial \eta}\left(a_{22}\frac{\partial \theta}{\partial \eta}\right) - $$
$$- J\mu_\theta \frac{\partial}{\partial \xi}\left(a_{12}\frac{\partial \theta}{\partial \eta}\right) - J\mu_\theta \frac{\partial}{\partial \eta}\left(a_{12}\frac{\partial \theta}{\partial \xi}\right) \tag{22}$$

where

$$a_1 = uJy_\eta - \nu Jx_\eta, a_2 = -uJy_\xi + \nu Jx_\xi, a_{11} = J(y_\eta^2 + x_\eta^2),$$

$$a_{12} = J(y_\xi y_\eta + x_\xi x_\eta), a_{22} = J(y_\xi^2 + x_\xi^2),$$

$J = \frac{1}{x_\xi y_\eta - x_\eta y_{xi}}$ is the Jacobian of transformation.

The use of transformation into a curvilinear coordinate system allows us to consider the problem (20)-(22) on a uniform rectangular grid and obtain qualitative pictures of simulated processes at moderate amounts of grid nodes. A metric conversion factors can be calculated analytically or numerically, depending on the definition or specification of the curvilinear boundary. An explicit scheme and an iterative method of successive upper relaxation is used to solve the problem numerically. The differential problem is replaced by a difference analogue of the following form:

$$\frac{\omega_{i,j}^{n+1} - \omega_{i,j}^n}{\tau} + \Lambda_{1,h}\omega_{i,j}^n + \Lambda_{2,h}\omega_{i,j}^n = \mu_{u,i,j}\Lambda_{11,h}\omega_{i,j}^n +$$
$$\mu_{u,i,j}\Lambda_{22,h}\omega_{i,j}^n - \mu_{u,i,j}\Lambda_{12,h}\omega_{i,j}^n - \mu_{u,i,j}\Lambda_{21,h}\omega_{i,j}^n + \beta_{i,j}\Phi_h\Theta_{i,j}^n \tag{23}$$

$$\Lambda_{11,h}\psi_{i,j}^{n+1} + \Lambda_{11,h}\psi_{i,j}^{n+1} - \Lambda_{12,h}\psi_{i,j}^{n+1} - \Lambda_{21,h}\psi_{i,j}^{n+1} = \frac{\omega_{i,j}^{n+1}}{J_{i,j}} \tag{24}$$

$$u_{i,j}^{n+1} = J_{i,j}\left[-x_{\eta,i,j}\frac{\psi_{i+1,j}^{n+1} - \psi_{i-1,j}^{n+1}}{2h_1} + x_{\xi,i,j}\frac{\psi_{i,j+1}^{n+1} - \psi_{i,j-1}^{n+1}}{2h_2}\right] \tag{25}$$

$$\nu_{i,j}^{n+1} = J_{i,j}\left[-y_{\eta,i,j}\frac{\psi_{i+1,j}^{n+1} - \psi_{i-1,j}^{n+1}}{2h_1} + y_{\xi,i,j}\frac{\psi_{i,j+1}^{n+1} - \psi_{i,j-1}^{n+1}}{2h_2}\right] \tag{26}$$

$$\frac{\theta_{i,j}^{n+1} - \theta_{i,j}^n}{\tau} + \Lambda_{1,h}\theta_{i,j}^n + \Lambda_{2,h}\theta_{i,j}^n =$$
$$\mu_{\theta,i,j}\Lambda_{11,h}\theta_{i,j}^n + \mu_{\theta,i,j}\Lambda_{22,h}\theta_{i,j}^n - \mu_{\theta,i,j}\Lambda_{12,h}\theta_{i,j}^n - \mu_{\theta,i,j}\Lambda_{21,h}\theta_{i,j}^n \tag{27}$$

where

$$a_{1,i,j} = J_{i,j}(y_{\eta,i,j}u_{i,j}^n - x_{\eta,i,j}\nu_{i,j}^n), a_{2,i,j} = J_{i,j}(x_{\xi,i,j}\nu_{i,j}^n - y_{\xi,i,j}u_{i,j}^n),$$

$$a_{11,i,j} = J_{i,j}(y_{\eta,i,j}^2 + x_{\eta,i,j}^2), a_{22} = J_{i,j}(y_{\xi,i,j}^2 + x_{\xi,i,j}^2),$$

$$a_{12,i,j} = a_{21,i,j} = J_{i,j}(y_{\xi,i,j}y_{\eta,i,j} + x_{\xi,i,j}x_{\eta,i,j}).$$

The difference analogues of the corresponding differential operators have the following form:

$$\Lambda_{1,h}\omega_{i,j}^n = \frac{1}{2}\left[\begin{array}{l}(a_{1,i+1/2,j} - |a_{1,i+1/2,j}|)\frac{\omega_{i+1,j}^n - \omega_{i,j}^n}{h_1} + \\ (a_{1,i-1/2,j} + |a_{1,i-1/2,j}|)\frac{\omega_{i,j}^n - \omega_{i-1,j}^n}{h_1}\end{array}\right],$$

$$\Lambda_{2,h}\omega_{i,j}^n = \frac{1}{2}\left[\begin{array}{c}(a_{2,i,j+1/2} - |a_{2,i,j+1/2}|)\frac{\omega_{i,j+1}^n - \omega_{i,j}^n}{h_2} + \\ (a_{2,i,j-1/2} + |a_{2,i,j-1/2}|)\frac{\omega_{i,j}^n - \omega_{i,j-1}^n}{h_2}\end{array}\right],$$

$$\Lambda_{11,h}\omega_{i,j}^n = \frac{J_{i,j}}{h_1}\left(a_{11,i+1/2,j}\frac{\omega_{i+1,j}^n - \omega_{i,j}^n}{h_1} - a_{11,i-1/2,j}\frac{\omega_{i,j}^n - \omega_{i-1,j}^n}{h_1}\right),$$

$$\Lambda_{22,h}\omega_{i,j}^n = \frac{J_{i,j}}{h_2}\left(a_{22,i,j+1/2}\frac{\omega_{i,j+1}^n - \omega_{i,j}^n}{h_2} - a_{22,i,j-1/2}\frac{\omega_{i,j}^n - \omega_{i,j-1}^n}{h_2}\right),$$

$$\Lambda_{12,h}\omega_{i,j}^n = \frac{J_{i,j}}{2h_1}\left(a_{12,i+1,j}\frac{\omega_{i+1,j+1}^n - \omega_{i+1,j-1}^n}{2h_2} - a_{12,i-1,j}\frac{\omega_{i-1,j+1}^n - \omega_{i-1,j-1}^n}{2h_2}\right),$$

$$\Lambda_{21,h}\omega_{i,j}^n = \frac{J_{i,j}}{2h_2}\left(a_{12,i,j+1}\frac{\omega_{i+1,j+1}^n - \omega_{i-1,j+1}^n}{2h_1} - a_{12,i,j-1}\frac{\omega_{i+1,j-1}^n - \omega_{i-1,j-1}^n}{2h_1}\right),$$

$$\Phi_h\theta_{i,j}^n = J_{i,j}\left[y_{\eta,i,j}\frac{\theta_{i+1,j}^n - \theta_{i-1,j}^n}{2h_1} - y_{\xi,i,j}\frac{\theta_{i,j+1}^n - \theta_{i,j-1}^n}{2h_2}\right].$$

The algorithm of numerical implementation is carried out in the following way: first of all, $\omega_{i,j}^{n+1}$ is found by (23); then $\psi_{i,j}^{n+1}$ is found by (24); $u_{i,j}^{n+1}$ and $\nu_{i,j}^{n+1}$ are determined from (25), (26) using the found values of $\psi_{i,j}^{n+1}$; the values of $\theta_{i,j}^{n+1}$ are calculated using the new values of $u_{i,j}^{n+1}$ and $\nu_{i,j}^{n+1}$ from (27). The iteration process continued until the following condition is met:

$$\max_{\binom{1\le i\le n_1}{1\le j\le n_2}}|\omega_{i,j}^{n+1} - \omega_{i,j}^n| \le \varepsilon.$$

The use of an explicit scheme and slowly convergent iterative methods is explained by the fact that rapidly convergent cost-effective methods, using implicit schemes, require self-adjointness and positive definiteness of the matrix of differential operators. This complicates the problem in the presence of the metric tensors coefficients. According to the algorithm described above, numerous methodological calculations are carried out in various doubly connected domains. Dimensionless quantities of velocity, length, temperature, and time were used in the calculations. In the example, the temperature was assumed to be $\theta = 0$ in the outer boundary, and $\theta = 1$ in the inner boundary. A ring was specially chosen in the first example, where the boundaries are described by the equations of a circle, the grid is constructed by an algebraic method, and the components of the metric tensor are determined analytically (Fig. 10a). Figure 10b shows the results of calculating the temperature change. It can be seen from the figure that the cold liquid is based in the lower part of the region, since it is known from the physics course that the density of the liquid is inversely proportional to the temperature.

Since the process is not stationary, the norms of velocity and temperature also do not tend to a stationary regime in calculations. According to this, the vortex regimes constantly change and the cooler liquid swings in the lower part of the calculated domain. Figure 11 shows the results of calculating the temperature and the current function at the same time.
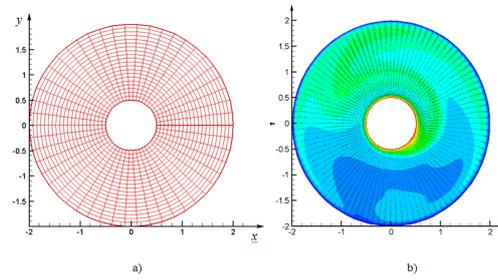
Figure 8. Estimated grid and the results of numerical solution of the temperature change
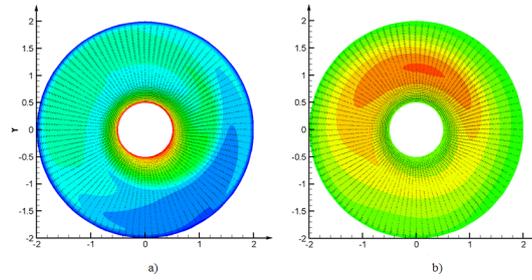
Figure 9. The results of a numerical solution of a) temperature change, b) the current function at the same time
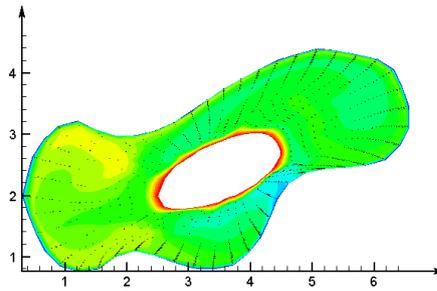
Figure 10. Results of the numerical solution of the temperature variation in the curvilinear domain
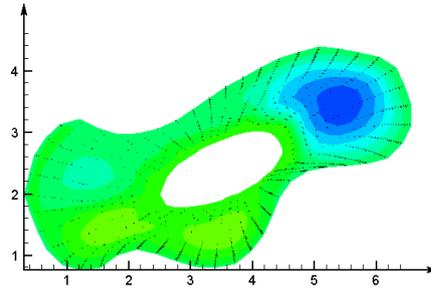
Figure 11. Results of the numerical solution of the current function in the curvilinear domain

Figures 12 and 13 show the results of numerical calculations of the temperature and current functions at the same time. It is seen from the figures that vortex motions are formed in the curves of the boundary, and a warm liquid accumulates in the upper parts of the bend. Since the process is non-stationary, the modes of vortices are constantly changing, and the temperature of the fluid is constantly transferred. In conclusion, it can be seen that the construction of a curvilinear grid for the description of convective flow helps to obtain a qualitative description of physical processes.

Conclusion

Thus, a method for the numerical construction of curvilinear structured grids in doubly connected domains in studied in this paper. Numerical modeling of the convective flow of a nonuniformly heated liquid in a curvilinear coordinate system is performed. The implicit scheme and the method of fractional steps are used in the numerical construction of curvilinear grids in doubly connected domains by equidistribution methods and the method of Godunov-Thompson. An explicit scheme and the method of fractional steps are used in the numerical implementation of the equations of an incompressible fluid. A cyclic run is used in the direction of the outer and inner boundaries, and a scalar run is used in the normal direction. Calculations were carried out for different cavity configurations and temperature regimes at the boundary. The plots of numerical calculations of the temperature and current function are obtained.

References

1. Thompson, J.F., Warsi, Z.U.A., Mastin, C.W.: Numerical grid generation, foundations and applications. New York, Elsevier (1985)
2. Shokin, Y.I., Danaev, N.T., Khakimzyanov, G.S., Shokina, N.Yu.: Lectures on difference schemes on moving grids. Almaty (2008)

3. Thompson, J.F., Warsi, Z.U.A., Mastin, C.W.: Numerical grid generation, foundations and applications. New York, Elsevier (1985)
4. Godunov, S.K., Zabrodin, A.V., Ivanov, M.Ya.: Numerical solution of multidimensional problems of gas dynamics. Moskow, Nauka (1976)
5. Sirochenko, V.P.: Numerical modeling of convective viscous fluid flows in multiply connected regions: Proceedings of the International Conference RDAMM.: Spec. Issue T.6, Part 2, pp. 554-562 (2001)
6. Prokopov, G.P.: On the organization of comparison of algorithms and programs for constructing regular two-dimensional difference grids. Preprint No. 18. USSR Academy of Sciences, Keldysh Institute of Economics, Moskow, 27 pp., (1989)

A sufficient condition for pre-compactness of set on the global Morrey-type space

D. Matin and Zh. Baituyakova

L. N. Gumilev Eurasian National University, Astana, Kazakhstan

Abstract This paper is dedicated a sufficient condition for pre-compactness of setin the global Morrey-type space $GM_{p\theta}^w$. From the proved theorem for generalized Morrey space M_p^w [3], [4], [5], local Morrey-type spaces [6] andin the case of L_p this is well-known Frechet-Kolmogorov theorem.

Keywords: Morrey spaces, generalized Morrey spaces, precompactness, completely boundedness

Introduction

Let $1 \leq p \leq \infty$, w be a measurable nonnegative function on $(0, \infty)$ that is not equivalent to zero. A global space of Morrey type $GM_{p\theta}^{w(\cdot)}$ $equivGM_{p\theta}^{w(\cdot)}(\mathbb{R}^n)$ is defined as the set of all functions $f \in L_p^{loc}(\mathbb{R}^n)$ with a finite quasinorm

$$\|f\|_{GM_{p\theta}^{w(\cdot)}} \equiv \sup_{x \in \mathbb{R}^n, \ r>0} \left\| w(r) \|f\|_{L_p(B(x,r))} \right\|_{L_\theta(0,\infty)},$$

where $B(x, r)$ is an open ball with center at the point $x \in \mathbb{R}^n$ of radius $r > 0$.

The space $GM_{p\theta}^{w(\cdot)}$ coincides with the generalized Morrey space for $\theta = \infty$ [7], [8]. The generalized Morrey space $M_p^{w(\cdot)}$ coincides with the classical Morrey space M_p^λ for $w(r) = r^{-\lambda}$, where $0 \leq \lambda \leq \frac{n}{p}$, which,in turn, when $\lambda = 0$ coincides with the space $L_p(\mathbb{R}^n)$, and for $\lambda = \frac{n}{p}$ with $L_\infty(\mathbb{R}^n)$.

In accordance with [1] and [2], we denote by $\Omega_{p\theta}$ the set of all functions that are nonnegative, measurable on $(0, \infty)$, not equivalent to 0 and such that for some $t > 0$ (and hence for any $t > 0$)

$$|w(r)r^{\frac{n}{p}} |_{L_\theta(0,t)} < \infty, \quad |w(r) |_{L_\theta(t,\infty)} < \infty.$$

The space $M_p^{w(\cdot)}$ is nontrivial, that is, it consists not only of functions equivalent to 0 on $\mathbb{R}^n$ if and only if $w \in \Omega_{p\theta}$ (see [1], [2]). Let $\chi(A)$ is the characteristic function of the set $A \subset \mathbb{R}^n$ and ^cA is the complement of A.

Theorem. Let $1 \leq p \leq \theta \leq \infty$ and $w \in \Omega_{p\theta}$.Suppose that the set $S \subset GM_{p\theta,w}(R^n)$ satisfies the following conditions:

$$\sup_{f \in S} \|f\|_{GM_{p\theta,w}} < \infty, \tag{1}$$

$$\lim_{\substack{u \to 0 \\ f \in S}} \sup \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}} = 0 \qquad (2)$$

and

$$\lim_{\substack{r \to \infty \\ f \in S}} \sup \left\| f \chi_{C_{B(0,r)}} \right\|_{GM_{p\theta,w}} = 0 \qquad (3)$$

Then S is a precompact set in $GM_{p\theta,w}(R^n)$.

In the case of the Morrey space $M_p^{\lambda}(0 < \lambda < \frac{n}{p})$, this theorem was proved in [5]-[13], and in the case $\lambda = 0$ - is the well-known Frechet-Kolmogorov theorem (see [14]).

To prove this theorem we need the following auxiliary assertions.

For $f \in L_1^{loc}(\mathbb{R}^n)$ and $r > 0$, we denote by

$$(M_r f)(x) = \frac{1}{|B(x,r)|} \int_{B(x,r)} f(y) dy,$$

where $mid A \ mid$ denotes the Lebesgue measure of the set $A \subset \mathbb{R}^n$.

Lemma 1. Let $1 \le p \le \theta \le \infty$, $w \in \Omega_\theta$. Then the following estimate holds for all $f \in LM_{p\theta}^{w(\cdot)}$ and $r > 0$:

$$\|M_r f - f\|_{LM_{p\theta,w}} \le \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{LM_{p\theta,w}} \qquad (4)$$

If $w \in \Omega_{p,\theta}$, then the following estimate holds for all $f \in GM_{p\theta}^{w(\cdot)}$ and $r > 0$:

$$\|M_r f - f\|_{GM_{p\theta,w}} \le \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}} \qquad (5)$$

Proof of the lemma 1. Let $z \in R^n$ и $\rho > 0$. Then, according to Holder's inequality

$$\|M_r f - f\|_{L_p(B(z,\rho))} =$$

$$= \left(\int_{B(z,\rho)} \left| \frac{1}{|B(x,r)|} \int_{B(x,r)} f(y) dy - f(x) \right|^p dx \right)^{\frac{1}{p}}$$

$$= \left(\int_{B(z,\rho)} \left| \frac{1}{|B(x,r)|} \int_{B(x,r)} (f(y) - f(x)) dy \right|^p dx \right)^{\frac{1}{p}}$$

$$\le \left(\int_{B(z,\rho)} \left(\frac{1}{|B(x,r)|} \int_{B(x,r)} |f(y) - f(x)|^p dy \right) dx \right)^{\frac{1}{p}}$$

$(y = x + u)$

$$= \left(\int_{B(z,\rho)} \left(\frac{1}{|B(0,r)|} \int_{B(0,r)} |f(x+u) - f(x)|^p du \right) dx \right)^{\frac{1}{p}}$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\int\limits_{B(z,\rho)} |f(x+u) - f(x)|^p dx \right) du \right)^{\frac{1}{p}}$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \|f(\cdot + u) - f(\cdot)\|^p_{L_p(B(z,\rho))} \, du \right)^{\frac{1}{p}}$$

Next

$$\|M_r f - f\|_{LM_{p\theta,w}} = \left\| w(\rho) \, \|M_r f - f\|_{L_p(B(0,\rho))} \right\|_{L_\theta(0,\infty)}$$

$$\leq \left\| w(\rho) \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \|f(\cdot + u) - f(\cdot)\|^p_{L_p(B(0,\rho))} \, du \right)^{\frac{1}{p}} \right\|_{L_\theta(0,\infty)}$$

$$= \left\| \frac{1}{|B(0,r)|} \int\limits_{B(0,r)} w(\rho)^p \|f(\cdot + u) - f(\cdot)\|^p_{L_p(B(0,\rho))} \, du \right\|^{\frac{1}{p}}_{L_{\frac{\theta}{p}}(0,\infty)}$$

Since $\frac{\theta}{p} \geq 1$, then applying the Minkowski inequality for integrals, we get that

$$\|M_r f - f\|_{LM_{p\theta,w}} \leq \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\int\limits_0^\infty w(\rho)^\theta \|f(\cdot + u) - f(\cdot)\|^\theta_{L_p(B(0,\rho))} \, d\rho \right)^{\frac{p}{\theta}} du \right)^{\frac{1}{p}}$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \|f(\cdot + u) - f(\cdot)\|^p_{LM_{p\theta,w}} \, du \right)^{\frac{1}{p}}$$

$$\leq \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{LM_{p\theta,w}} \quad (*)$$

Since

$$(M_r f(\cdot + z)) (x) = \frac{1}{|B(x,r)|} \int\limits_{B(x,r)} f(y+z) dy =$$

$$= \frac{1}{|B(x+z,r)|} \int\limits_{B(x+z,r)} f(u) du = (M_r f)(x+z),$$

then for global spaces $GM_{p\theta,w}$ we have, using $(*)$, that

$$\|M_r f - f\|_{GM_{p\theta,w}} = \sup_{z \in R^n} \|(M_r f)(\cdot + z) - f(\cdot + z)\|_{LM_{p\theta,w}}$$

$$= \sup_{z \in R^n} \|M_r(f(\cdot + z)) - f(\cdot + z)\|_{LM_{p\theta,w}}$$

$$\leq \sup_{z \in R^n} \sup_{u \in B(0,r)} \|f(\cdot + u + z) - f(\cdot + z)\|_{LM_{p\theta,w}}$$

$$= \sup_{z \in R^n} \sup_{u \in B(0,r)} \left\| w(\rho) \|f(\cdot + u + z) - f(\cdot + z)\|_{L_p(B(0,\rho))} \right\|_{L_\theta(0,\infty)}$$

$$= \sup_{z \in R^n} \sup_{u \in B(0,r)} \left\| w(\rho) \|f(\cdot + u) - f(\cdot)\|_{L_p(B(z,\rho))} \right\|_{L_\theta(0,\infty)}$$

$$= \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}}.$$

Lemma 1 is proved.

Lemma 2. Let $1 \leq p \leq \theta \leq \infty$, $w \in \Omega_{p\theta}$. Then for all $f \in GM_{p\theta}^{w(\cdot)}$ and $r > 0$ the following inequality holds:

$$\|M_r f\|_{GM_{p\theta,w}} \leq \|f\|_{GM_{p\theta,w}} \tag{6}$$

Proof of the lemma 2. Then, according to Holder's inequality

$$\|M_r f\|_{L_p(B(z,\rho))} = \left(\int\limits_{B(z,\rho)} \left| \frac{1}{|B(x,r)|} \int\limits_{B(x,r)} f(y)dy \right|^p dx \right)^{\frac{1}{p}}$$

$$\leq \left(\int\limits_{B(z,\rho)} \left(\frac{1}{|B(x,r)|} \int\limits_{B(x,r)} |f(y)|^p \, dy \right) dx \right)^{\frac{1}{p}}$$

$$= (y = x + u)$$

$$= \left(\int\limits_{B(z,\rho)} \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} |f(x+u)|^p \, du \right) dx \right)^{\frac{1}{p}}$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\int\limits_{B(z,\rho)} |f(x+u)|^p \, dx \right) du \right)^{\frac{1}{p}}$$

$$= (x + u = v)$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\int\limits_{B(z+u,\rho)} |f(v)|^p \, dv \right) du \right)^{\frac{1}{p}}$$

$$= \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \|f\|_{L_p(B(z+u,\rho))}^p \, du \right)^{\frac{1}{p}}$$

Since $\frac{\theta}{p} \geq 1$, then applying the Minkowski inequality for integrals, we get that

$$
\|M_r f\|_{GM_{p\theta,w}} \leq \sup_{z \in R^n} \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\int\limits_0^\infty w(\rho)^\theta \|f\|_{L_p(B(z+u,\rho))}^\theta \, d\rho \right)^{\frac{p}{\theta}} du \right)^{\frac{1}{p}}
$$

$$
\leq \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \sup_{z \in R^n} \left(\int\limits_0^\infty w(\rho)^\theta \|f\|_{L_p(B(z+u,\rho))}^\theta \, d\rho \right)^{\frac{p}{\theta}} du \right)^{\frac{1}{p}}
$$

$$
= \sup_{z \in R^n} \left\| \frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(w(\rho) \|f\|_{L_p(B(z+u,\rho))} \right)^p du \right\|_{L_{\frac{\theta}{p}}(0,\infty)}^{\frac{1}{p}}
$$

$$
\leq \sup_{z \in R^n} \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left\| w(\rho) \|f\|_{L_p(B(z+u,\rho))} \right\|_{L_\theta(0,\infty)}^p du \right)^{\frac{1}{p}}
$$

$$
\leq \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \left(\sup_{v \in R^n} \left\| w(\rho) \|f\|_{L_p(B(v,\rho))} \right\|_{L_\theta(0,\infty)} \right)^p dv \right)^{\frac{1}{p}}
$$

$$
= \|f\|_{GM_{p\theta,w}}
$$

For $delta > 0$, let $w_{delta}(f, G)$ be the modulus of continuity of the function f on the set $G \subset R^n$:

$$
w_\delta(f, G) = \sup_{\substack{x_1, x_2 \in G \\ |x_1 - x_2| \leq \delta}} |f(x_1) - f(x_2)| .
$$

$$
\|M_r f\|_{GM_{p\theta,w}} = \sup_{z \in R^n} \left\| w(\rho) \|M_r f\|_{L_p(B(z,\rho))} \right\|_{L_\theta(0,\infty)}
$$

$$
\leq \sup_{z \in R^n} \left\| w(\rho) \left(\frac{1}{|B(0,r)|} \int\limits_{B(0,r)} \|f\|_{L_p(B(z+u,\rho))}^p du \right)^{\frac{1}{p}} \right\|_{L_\theta(0,\infty)}
$$

Lemma 2 is proved.

Lemma 3. Let $1 \leq p, \theta \leq \infty$, $w \in \Omega_{p\theta}$. Then there exists $r_0 > 0$ and for any $0 < r \leq r_0$ there exists $C_r > 0$ depending only on r, n, p, θ, w, what

1) for any $f \in GM_{p\theta}^{w(\cdot)}$

$$
\|M_r f\|_{C(R^n)} \leq C_r \|f\|_{GM_{p\theta,w}} . \tag{7}
$$

2)for all $\delta > 0$

$$w_\delta(M_r f; R^n) \le C_r \sup_{\substack{|u| \le \delta \\ u \in B(0,\delta)}} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}} . \tag{8}$$

Proof of the lemma 3. **1.** Since the function $w \in \Omega_{p\theta}$ is not equivalent to 0, then there exists $r_0 > 0$ such that $\|w\|_{L_\theta(r_0,\infty)} > 0$. Let $0 < r \le r_0$. By the Holder inequality for any $x \in R^n$

$$|M_r f(x)| \le \frac{1}{|B(x,r)|^{\frac{1}{p}}} \|f\|_{L_p(B(x,r))} .$$

Hence

$$\|w(\rho) M_r f(x)\|_{L_\theta(r,\infty)} \le \frac{1}{(v_n r^n)^{\frac{1}{p}}} \left\| w(\rho) \|f\|_{L_p(B(x,r))} \right\|_{L_\theta(r,\infty)} ,$$

where v_n is the volume of the unit ballin R^n, and

$$|M_r f(x)| \|w(\rho)\|_{L_\theta(r,\infty)} \le \frac{1}{(v_n r^n)^{\frac{1}{p}}} \left\| w(\rho) \|f\|_{L_p(B(x,r))} \right\|_{L_\theta(0,\infty)} ,$$

therefore

$$\sup_{x \in R^n} |M_r f(x)| \le C_r \sup_{x \in R^n} \left\| w(\rho) \|f\|_{L_p(B(x,r))} \right\|_{L_\theta(0,\infty)} = C_r \|f\|_{GM_{p\theta,w}} , \tag{9}$$

where $C_r = \left(\|w\|_{L_\theta(r,\infty)} (v_n r^n)^{\frac{1}{p}} \right)^{-1}$.
   2. For all $x_1, x_2 \in B(0,r)$

$$|(M_r f)(x_1) - (M_r f)(x_2)| = \frac{1}{v_n r^n} \left| \int\limits_{B(x_1,r)} f(y) dy - \int\limits_{B(x_2,r)} f(y) dy \right| =$$

$$= (v_n r^n)^{-1} \left| \int\limits_{B(0,r)} f(z + x_1) dz - \int\limits_{B(0,r)} f(z + x_2) dz \right| \le$$

$$\le (v_n r^n)^{-1} \int\limits_{B(0,r)} |f(z + x_1) - f(z + x_2)| dz =$$

$$= (v_n r^n)^{-1} \int\limits_{B(x_2,r)} |f(s + x_1 - x_2) - f(s)| ds \le$$

$$\le (v_n r^n)^{-\frac{1}{p}} \|f(\cdot + x_1 - x_2) - f(\cdot)\|_{L_p(B(x_2,r))}$$

Therefore, according to the first step of the proof

$$\sup_{\substack{x_1,x_2\in R^n \\ |x_1-x_2|\leq\delta}} |(M_rf)(x_1)-(M_rf)(x_2)| \leq C_r \sup_{\substack{x_1,x_2\in R^n \\ |x_1-x_2|\leq\delta}} \|f(\cdot+x_1-x_2)-f(\cdot)\|_{GM_{p\theta,w}} =$$

$$= C_r \sup_{\substack{|u|\leq\delta \\ u\in B(0,\delta)}} \|f(\cdot+u)-f(\cdot)\|_{GM_{p\theta,w}}$$

Lemma 3 is proved.

Lemma 4. Let $1\leq p,\theta\leq\infty$, $w\in\Omega_{p\theta}$. Then there exists $C>0$ depending only on n,p,θ,w such that for any $r,R>0$ and for any $f,g\in GM_{p\theta,w}$ we have the estimate

$$\|M_rf-M_rg\|_{GM_{p\theta,w}} \leq C\left(1+R^{\frac{n}{p}}\right)\|M_rf-M_rg\|_{C\left(\overline{B(0,R)}\right)} +$$

$$+ \sup_{u\in B(0,R)} \|f(\cdot+u)-f(\cdot)\|_{GM_{p\theta,w}} +$$

$$+ \sup_{u\in B(0,R)} \|g(\cdot+u)-g(\cdot)\|_{GM_{p\theta,w}} + \left\|f\chi_{{}^cB(0,R)}\right\|_{GM_{p\theta,w}} + \left\|g\chi_{{}^cB(0,R)}\right\|_{GM_{p\theta,w}}$$

Proof of the lemma 4. Really,

$$\|M_rf-M_rg\|_{GM_{p\theta,w}} \leq \left\|(M_rf-M_rg)\chi_{B(0,R)}\right\|_{GM_{p\theta,w}} +$$

$$+ \left\|(M_rf-M_rg)\chi_{{}^cB(0,R)}\right\|_{GM_{p\theta,w}} = I_1+I_2$$

Next

$$I_1 = \sup_{x\in R^n} \left\|w(\rho)\|M_rf-M_rg\|_{L_p(B(x,\rho)\cap B(0,R))}\right\|_{L_\theta(0,\infty)}$$

$$\leq \sup_{x\in R^n} \left\|w(\rho)\|M_rf-M_rg\|_{L_p(B(x,\rho)\cap B(0,R))}\right\|_{L_\theta(0,1)}$$

$$+ \sup_{x\in R^n} \left\|w(\rho)\|M_rf-M_rg\|_{L_p(B(x,\rho)\cap B(0,R))}\right\|_{L_\theta(1,\infty)}$$

$$\leq \|M_rf-M_rg\|_{C\left(\overline{B(0,R)}\right)} \cdot$$

$$\cdot\left(\left\|w(\rho)(v_n\rho^n)^{\frac{1}{p}}\right\|_{L_\theta(0,1)} + \left\|w(\rho)(v_nR^n)^{\frac{1}{p}}\right\|_{L_\theta(1,\infty)}\right)$$

$$\leq C\left(1+R^{\frac{n}{p}}\right)\|M_rf-M_rg\|_{C\left(\overline{B(0,R)}\right)}$$

where

$$C = v_n^{\frac{1}{p}}\left(\left\|w(\rho)\rho^{\frac{n}{p}}\right\|_{L_\theta(0,1)} + \|w(\rho)\|_{L_\theta(1,\infty)}\right) < \infty,$$

at $w\in\Omega_{p\theta}$.

In addition, according to Lemma 1

$$I_2 \leq \|M_rf-f\|_{GM_{p\theta,w}} + \left\|(f-g)\chi_{{}^cB(0,R)}\right\|_{GM_{p\theta,w}} + \|M_rg-g\|_{GM_{p\theta,w}}$$

$$\leq \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}} + \sup_{u \in B(0,r)} \|g(\cdot + u) - g(\cdot)\|_{GM_{p\theta,w}}$$

$$+ \left\| f\chi_{{}^{c}B(0,R)} \right\|_{GM_{p\theta,w}} + \left\| g\chi_{{}^{c}B(0,R)} \right\|_{GM_{p\theta,w}}$$

from which the desired inequality follows. Lemma 4 is proved.

Lemma 5. Let $1 \leq p, \theta \leq \infty$, $w \in \Omega_{p\theta}$. Then for any $r, R > 0$ and for any $f, g \in GM_{p\theta,w}$

$$\|f - g\|_{GM_{p\theta,w}} \leq C \left(1 + R^{\frac{n}{p}}\right) \|M_r f - M_r g\|_{C(\overline{B(0,R)})}$$

$$+2 \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta,w}} + 2 \sup_{u \in B(0,r)} \|g(\cdot + u) - g(\cdot)\|_{GM_{p\theta,w}} \qquad (10)$$

$$+ \left\| f\chi_{{}^{c}B(0,R)} \right\|_{GM_{p\theta,w}} + \left\| g\chi_{{}^{c}B(0,R)} \right\|_{GM_{p\theta,w}},$$

where $C > 0$ is the same asin Lemma 4.

Proof of the lemma 5. It suffices to note that

$$\|f - g\|_{GM_{p\theta,w}} \leq \|M_r f - f\|_{GM_{p\theta,w}} + \|M_r f - M_r g\|_{GM_{p\theta,w}} + \|M_r g - g\|_{GM_{p\theta,w}}$$

and use Lemmas 1 and 4

Proof of the theorem. Let $S \subset GM_{p\theta}^{w(\cdot)}$ and suppose that conditions (1) - (3) are satisfied.

Step 1. Let $0 < r < r_0$, where r_0 is definedin Lemma 3, and $R > 0$ are fixed. By virtue of inequality (7) and condition (1), it follows that

$$\sup_{f \in S} \|M_r f\|_{C(\overline{B(0,R)})} < \infty.$$

Moreover, by virtue of inequality (8) and condition (2), it follows that

$$\lim_{u \to 0} \sup_{f \in S} \|M_r f(\cdot + u) - M_r f(\cdot)\|_{C(\overline{B(0,R)})} = 0.$$

Consequently, by the Ascoli-Arzela theorem, the set $S_r = \{M_r f : f \in S\}$ is precompactin $C(\overline{B(0,R)})$, or, which is the same, set S_r is completely bounded, that is, for any $varepsilon > 0$ there exist $m \in \mathbb{N}$, $f_1, ..., f_m \in S$ (depending on $varepsilon, r$, and R) such that for any $f \in S$

$$\min_{j=1,...,m} \|M_r f - M_r f_j\|_{C(\overline{B(0,R)})} < \varepsilon.$$

Step 2. Let $\{\varphi_1, ..., \varphi_m\}$ be an arbitrary finiteubset of S. By virtue of inequality (10), for any $f \in S$ and any $j = 1, ..., m$

$$\|f - \varphi_j\|_{GM_{p\theta}^{w}} \leq C(1 + R^{\frac{n}{p}}) \|M_r f - M\varphi_j\|_{C(\overline{B(0,R)})}$$

$$+2 \sup_{u \in B(0,r)} \|f(\cdot + u) - f(\cdot)\|_{GM_{p\theta}^{w(\cdot)}} + 2 \sup_{u \in B(0,r)} \|\varphi_j(\cdot + u) - \varphi_j(\cdot)\|_{GM_{p\theta}^{w(\cdot)}}$$

$$+ \left\| f \chi_{{}^c B(0,R)} \right\|_{GM_{p\theta}^{w(\cdot)}} + \left\| \varphi_j \chi_{{}^c B(0,R)} \right\|_{GM_{p\theta}^{w(\cdot)}}$$

$$\leq C(1 + \mathbb{R}^{\frac{n}{p}}) \left\| M_r f - M_r \varphi_j \right\|_{C(\overline{B(0,R)})} +$$

$$+4 \sup_{u \in B(0,r)} \sup_{g \in S} \|g(\cdot + u) - g(\cdot)\|_{GM_{p\theta}^w} + 2 \sup_{g \in S} \left\| g \chi_{{}^c B(0,R)} \right\|_{GM_{p\theta}^w}.$$

Hence,

$$\min_{j=1,...,m} \|f - \varphi_j\|_{GM_{p\theta}^w} \leq C(1 + \mathbb{R}^{\frac{n}{p}}) \min_{j=1,...,m} \left\| M_r f - M_r \varphi_j \right\|_{C(\overline{B(0,R)})} +$$

$$+4 \sup_{u \in B(0,r)} \sup_{g \in S} \|g(\cdot + u) - g(\cdot)\|_{GM_{p\theta}^w} + 2 \sup_{g \in S} \left\| g \chi_{{}^c B(0,R)} \right\|_{GM_{p\theta}^w}. \qquad (11)$$

Step 3. Let $\varepsilon > 0$. First, using condition (3), we find a $R(\varepsilon) > 0$ such that

$$\sup_{g \in S} \left\| g \chi_{{}^c B(0,R(\varepsilon))} \right\|_{GM_{p\theta}^w} < \frac{\varepsilon}{6}.$$

Further, using condition (2), we find a $r(\varepsilon)$ such that

$$\sup_{u \in B(0,r(\varepsilon))} \sup_{g \in S} \|g(\cdot + u) - g(\cdot)\|_{GM_{p\theta}^w} < \frac{\varepsilon}{12}.$$

Finally, because of the precompactness of the set $S_{r(\varepsilon)}$ in $C(\overline{B(0,R(\varepsilon))})$ there exist $m(\varepsilon) \in \mathbb{N}$ and $f_{1,\varepsilon}, ..., f_{m(\varepsilon),\varepsilon} \in S$, for any $f \in S$

$$\min_{j=1,...,m(\varepsilon)} \left\| M_{r(\varepsilon)} f - M_{r(\varepsilon)} f_{j,\varepsilon} \right\|_{C(\overline{B(0,R(\varepsilon))})} < \frac{\varepsilon}{3C(1 + R(\varepsilon)^{\frac{n}{p}})}.$$

Consequently, by virtue of inequality (11) с $\varphi_j = f_{j,\varepsilon}, j = 1, ..., m(\varepsilon)$, for any $f \in S$

$$\min_{j=1,...,m(\varepsilon)} \|f - f_{j,\varepsilon}\|_{GM_{p\theta}^w} < \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon.$$

This means that the set S is completely boundedin $GM_{p\theta}^w$, or, what is the same, the set S is precompactin $GM_{p\theta}^w$, which completes the proof of the theorem.

Remark. Condition (1) in the theorem is necessary, since any precompact setin a normed space is bounded.

As regards conditions (2) and (3), they are not necessary at any rate for $n = 1$ and $w(r) = r^{-\lambda}$, $0 < \lambda < \frac{1}{p}$, since the set S consisting of only one function $|x|^{\lambda - \frac{1}{p}} \in M_p^\lambda$ is precompact, but Conditions (2) and (3) are not satisfied. This follows from the example below.

Thus, the question of finding the necessary and sufficient conditions for the precompactness of the set $S \subset GM_{p\theta}^{w(\cdot)}$ remains open. *Example.* When $n = 1$ и $w(r) = r^{-\lambda}$, $1 \leq p < \infty$, $0 < \lambda < \frac{1}{p}$,

$$\left(M_r(|\cdot|^{\lambda - \frac{1}{p}}) \right)(x) \nrightarrow |x|^{\lambda - \frac{1}{p}} \qquad (12)$$

в M_p^λ at $r \to 0^+$,

$$|x + u|^{\lambda - \frac{1}{p}} \nrightarrow |x|^{\lambda - \frac{1}{p}} \tag{13}$$

в M_p^λ at $u \to 0$,

$$|x|^{\lambda - \frac{1}{p}} \chi_{\underset{B(0,r)}{c}}(x) \nrightarrow 0 \tag{14}$$

в M_p^λ at $r \to +\infty$,

Indeed, for $x > 0$ и $0 < r < x$

$$M_r(|\cdot|^{\lambda - \frac{1}{p}})(x) = \frac{1}{2r} \int\limits_{x-r}^{x+r} y^{\lambda - \frac{1}{p}} dy$$

$$= \left(\lambda - \frac{1}{p} + 1\right)^{-1} \frac{1}{2r} \left((x + r)^{\lambda - \frac{1}{p} + 1} - (x - r)^{\lambda - \frac{1}{p} + 1}\right)$$

$$= \left(\lambda - \frac{1}{p} + 1\right)^{-1} \frac{x^{\lambda - \frac{1}{p} + 1}}{2r} \left[\left(1 + \frac{r}{x}\right)^{\lambda - \frac{1}{p} + 1} - \left(1 - \frac{r}{x}\right)^{\lambda - \frac{1}{p} + 1}\right]$$

and

$$M_r(|\cdot|^{\lambda - \frac{1}{p}})(x) - x^{\lambda - \frac{1}{p}}$$

$$= \left(\lambda - \frac{1}{p} + 1\right)^{-1} \frac{x^{\lambda - \frac{1}{p} + 1}}{2r} \left[\left(1 + \frac{r}{x}\right)^{\lambda - \frac{1}{p} + 1} - \left(1 - \frac{r}{x}\right)^{\lambda - \frac{1}{p} + 1} - 2\left(\lambda - \frac{1}{p} + 1\right)\frac{r}{x}\right]$$

Making use of inequality [1]

$$(1 + y)^\mu - (1 - y)^\mu - 2\mu y \geq \frac{\mu(1 - \mu)(2 - \mu)}{3} y^3$$

which is valid for any $0 < mu < 1$ and $0 < y < 1$, we obtain by setting $mu = \lambda - \frac{1}{p} + 1$ and $y = \frac{r}{x}$ such that for some $c > 0$ depending only on λ and p, for any $0 < r < x$

$$M_r(|\cdot|^{\lambda - \frac{1}{p}})(x) - x^{\lambda - \frac{1}{p}} \geq cx^{\lambda - \frac{1}{p}} \left(\frac{r}{x}\right)^2.$$

---

[1]   Indeed, according to the Taylor formula, there exist ξ, η that $1 - x < \eta < 1 < \xi < 1 + x$ and

$$(1 + x)^\mu - (1 - x)^\mu - 2\mu x = 1 + \mu x + \frac{\mu(\mu - 1)}{2} x^2 + \frac{\mu(\mu - 1)(\mu - 2)}{6} x^3 +$$

$$+ \frac{\mu(\mu - 1)(\mu - 2)(\mu - 3)}{24} \xi^{\mu - 4} x^4$$

$$- \left(1 - \mu x + \frac{\mu(\mu - 1)}{2} x^2 - \frac{\mu(\mu - 1)(\mu - 2)}{6} x^3 + \frac{\mu(\mu - 1)(\mu - 2)(\mu - 3)}{24} \eta^{\mu - 4} x^4\right) - 2\mu x$$

$$= \frac{\mu(\mu - 1)(\mu - 2)}{3} x^3 + \frac{\mu(\mu - 1)(\mu - 2)(\mu - 3)}{24} \left(\xi^{\mu - 4} - \eta^{\mu - 4}\right) x^4 \geq \frac{\mu(\mu - 1)(\mu - 2)}{3} x^3.$$

Hence

$$\left\| M_r(|\cdot|^{\lambda-\frac{1}{p}})(x) - |x|^{\lambda-\frac{1}{p}} \right\|_{M_p^\lambda} = \sup_{z\in\mathbb{R},\ r>0} r^{-\lambda} \left\| M_r(|\cdot|^{\lambda-\frac{1}{p}})(x) - |x|^{\lambda-\frac{1}{p}} \right\|_{L_p(z-r,z+r)}$$

$$\geq r^{-\lambda} \left\| M_r(|\cdot|^{\lambda-\frac{1}{p}})(x) - |x|^{\lambda-\frac{1}{p}} \right\|_{L_p(2r,4r)}$$

$$\geq cr^{-\lambda} \left\| x^{\lambda-\frac{1}{p}} \left(\frac{r}{x}\right)^2 \right\|_{L_p(2r,4r)}$$

$$\geq c\ r^{-\lambda}(4r)^{\lambda-\frac{1}{p}} \left(\frac{1}{4}\right)^2 (2r)^{\frac{1}{p}} = c\ 4^{\lambda-\frac{1}{p}-2} 2^{\frac{1}{p}} > 0,$$

from which (12) follows. From (12), according to Lemma 1, (13) follows.

Finally

$$\left\| |x|^{\lambda-\frac{1}{p}} \chi_{\underset{B(0,r)}{c}}(x) \right\|_{M_p^\lambda} = \sup_{z\in\mathbb{R}^n,\ \rho>0} \left\| |x|^{\lambda-\frac{1}{p}} \chi_{\underset{B(0,r)}{c}}(x) \right\|_{L_p(x-\rho,x+\rho)}$$

$$\geq \sup_{\rho>r} \rho^{-\lambda} \left\| |x|^{\lambda-\frac{1}{p}} \right\|_{L_p((0,\rho)\cap(r,\infty))} = \sup_{\rho>r} \rho^{-\lambda} \left(nv_n \int_r^\rho x^{\lambda p-1} dx \right)^{\frac{1}{p}}$$

$$\geq \lim_{\rho\to+\infty} \rho^{-\lambda} \left(\frac{nv_n}{\lambda p} \left(\rho^{\lambda p-1} - r^{\lambda p-1} \right) dx \right)^{\frac{1}{p}} = \left(\frac{nv_n}{\lambda p} \right)^{\frac{1}{p}},$$

from which (14) follows.

References

1. Burenkov V. I. Recent progress in studying the boundedness of classical operators of real analysis in general Morrey-type spaces. I.   *Eurasian Mathematical Journal*, 2012, Volume 3, Number 3, pp. 11 - 32.
2. Burenkov V. I. Recent progress in studying the boundedness of classical operators of real analysis in general Morrey-type spaces. II. *Eurasian Mathematical Journal*, 2013, Volume 4, Number 1, pp. 21 - 45.
3. Chen Y., Ding Y., Wang X. , Compactness of commutators of Riesz potential on Morrey space., *Potential Anal.*, 2009, Volume 30. 4, pp. 301 – 313.
4. Chen Y., Ding Y., Wang X. Compactness of Commutators for singular integrals on Morrey spaces. *Canad. J. Math.*, 2012, Volume 64(2), pp. 257-281.
5. Guliyev V. S. , Generalized weighted Morrey spaces and higher order commutators of sublinear operators., *Eurasian Mathematical Journal*, 2012, Volume 3. 3, pp. 33 – 61.
6. Bokayev N.A., Matin D.T. Compactness of Commutators for one type of singular integrals on generalized Morrey spaces.   *Springer Proceedings in Mathematics Statistics*, 2017, Volume 216, pp. 47-54.
7. Bokayev N.A., Matin D.T. A sufficient of compactness of the commutators of Riesz potential on global Morrey-type space.   *Herald ENU*, 2017, Number 6(121), pp. 18-24.

8. Bokayev N., Burenkov V., Matin D.A sufficient condition for compactness of commutators for Riesz potential on the generalized Morrey space *Herald ENU*, 2016, Number 6(115), pp. 8-13.

9. Bokayev N., Burenkov V., Matin D. On the pre-compactness of a set in the generalized Morrey spaces. *AIP Conference Proceedings*, 2016, Volume 1759, pp. 020108-01–03.doi: 10.1063/1.4959722

10. Bokayev N.A., Burenkov V.I., Matin D.T. , Sufficient conditions for pre-compactness of sets in the generalized Morrey spaces. *Bulletin of the KARAGANDA UNIVERCITY*, 2016, Number 4, pp. 18—40.

11. Bokayev N., Burenkov V., Matin D. Sufficient conditions for the pre-compactness of sets in global Morrey-type spaces *AIP Conference Proceedings*, 2017, Volume 1980, pp. 030001-01–06.doi: 10.1063/1.5000600

12. Matin D., Turgumbayev M.Zh. Sufficient conditions for the pre-compactness of sets in global Morrey-type spaces *Herald ENU*, 2017, Number 4(119), pp. 14-23.

13. Bokayev N., Burenkov V., Matin D. On precompactness of a set in general local and global Morrey-type spaces *Eurasian Mathematical Journal* , 2017, Volume 8, Number 3. pp.109 - 115.

14. Yosida K. Functional Analysis. Springer-Verlag, Berlin, 1978

Creation of the adaptive graphic Web interfaces for input and editing data for the heterogeneous information systems on the bases of XML technology

A. Mukhitova and O. Zhizhimov

Novosibirsk State University, Novosibirsk, Russia
Institute of Computational Technologies SB RAS, Novosibirsk, Russia

Abstract The work is devoted to an appliance of constructing adaptive technology of graphic administrative WEB-interface for the resolution of integration problems of heterogeneous information resources, on the ground of the XSD application database schemes with the appliance of XSLT-transformations. An approach is described for elaboration, as well as an example of implementation, of adaptive entry model and data editing in the way of created editor prototype of the XML-records. The described methodology has quite general ways of usage and can be applied for constructing adaptive graphic WEB-interfaces, enabling to generate shippable HTML-forms for entering and editing data.

Keywords: :adapting graphic administrative and users web-interfaces, integration of heterogeneous data, XML editor, SRU, XML, XSD, XSLT-transformations.

The paper deals [1] with the development of technologies for creating adaptive graphical interfaces for heterogeneous information systems. Heterogeneous distributed applications should consist of user and administrative interfaces (graphic WEB-interfaces) supplying an opportunity to govern data from various informational sources. Interfaces, adjusting to structure and functionality of informational resources, are adaptable. Each of a particular informational resource, in general, has quite narrow range of features from the potentially existing meanings. Therefore, attraction of additional data on a particular informational recourse is required, while choosing the component governing features for various informational resources.

The inputting information about functional qualities of each data sources is essential for implementation of adaptable interfaces in mentioned point of client interfaces. Depending on the technology access, used to obtain informational assets, that sort of data can be received and processed. The constructing adaptive

technology of client interfaces was considered in the work in order to manage the information search with the method of displaying extracted data using Z39.50 [1] and SRW/SRU [2],[3] on the basis of Explain services [3], [4], [5] in the various modifications. The implementation of these adaptable interfaces for the ZooSPACE [6],[7] platform was also illustrated. Constructed applications enable to automatically adjust graphic interfaces to possibilities of one or the other information resource:

- supported sets of searching attributes;
- supported combinations of searching attributes;
- supported schemes, patterns and sets of elements.

Despite the fact that there is a considerable number of web-resources and services, files of documents and libraries, which are applied as a data exchange format XML, the usage of JSON should be considered in constructing new distributed informational systems. At the present time, this format is developing rapidly and is easy for implementation in systems of data exchanging. For the reason of being simple and easy to operate data, JSON is better to use on the client's side, while the XML technologies are more convenient on the side of a server. In view of the specific character of the given problem the decision was made in favour of XML-technology.

The descriptions of the constructing rules of XML-records structure are essential in the use of XML format to present structured information, i.e. descriptions of data scheme. In general, the rules for XML are formulated in terms of XSD [8]-[10] and they present XML structure which can be processed with standard ways, for example XSLT [11].

The question about where the full description of potential structure of the derived record can be obtained arises in the process of extracting record from a particular informational source in heterogeneous informational system and presenting the record in the XML format for editing. The following options are possible [12]:

- The XML record, derived for editing, includes a reference on the applied XSD data scheme in the form of URL with schemaLocation as an attribute in determination of employed namespaces. It is usually contained in the XML record root element. In that case the issue of receiving XSD is solved in a trivial manner.
- The XML, record derived for editing, includes the namespace identification (URI), though it does not include a reference on the applied XSD data scheme in the form of URL. In that situation the informational system should be requested to provide the XSD in the use of namespace identification. For the ZooSPACE platform the similar request can be processed with Explain service.
- The XML record, derived for editing, does not include definitions of namespaces. In this case the informational system should be requested to provide the XSD (as a default) by the name of informational resource (database), or by using the XSD, which before corresponded to the scheme requested in an inquiry formation for extracting data.

In any of the listed cases the following steps are necessary for graphic interfaces initialisation of data modifications:

- The data scheme description in the way of the XML structure in accordance with the XSD rules.
- The XML structure consisting extracted data for editing (not required for creating a new record)
- The description of entity generation patterns of graphic interface (not necessary).
- The description of entity generation patterns of graphic interface in accordance with the XSD rules and with elements of editing the XML record.
- In these conditions, the XSLT conversion rules can be the rules applied to XSD.

The algorithm is presented in the Figure 1, as an illustration of work of the XML records prototype adaptive editor in the format of client-server, built in WEB server of the ZooSPACE (ZooSPACE-W) platform. The XML editor is relevant to an area restricted by a dashed line for server side. As for the client part, the beforehand prepared HTML form to input and/or edit data is provided. In these conditions, the form already contains the all needed tools (java scripts) for correct data input, which includes:

- A script for duplicating elements, the repetition of which is possible according to the XSD.
- A script for removing elements, providing that the removing is possible according to the XSD.
- A script for checking the accuracy of data entry, if there is a relevant pattern in the way of regular expression in the XSD.
- A script for hiding- revealing any data elements in the form of editing.

It should be taken into account that the XSD data scheme definitions can contain references to other XSD data scheme definitions, which complement definitions both in the current namespace (element xsd:include), and in the other namespaces (element xsd:import). Therefore the initial XSD structure, before being processed by the XSLT processor requires modifying to register extra definitions.

The editor of the XML records operating principle, in format of client-server built in WEB server, can be described as follows:

- as for the client part, the beforehand prepared HTML form to input and/or edit data is provided. In these conditions the form already contains the all needed tools (java scripts) for correct data input;
- generation of editing forms occurs on the server side with the XSLT method of transformations of the modified XSD structure. At the beginning of the process an empty editing form is produced (without data). As soon as the XSLT processor has completed its action, the empty form is filled with record data in XML format.
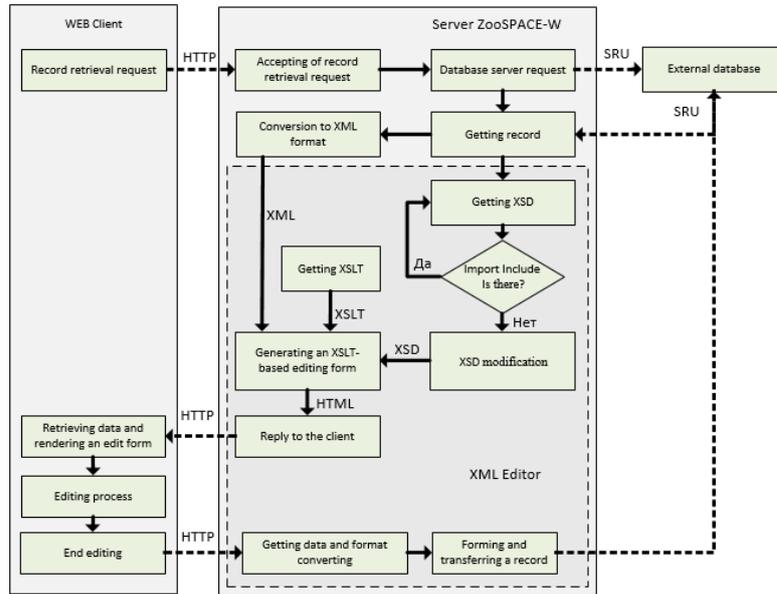
Figure 1. The work algorithm of editor process BD record in the heterogeneous system with example of the ZooSPACE-W

For generation of empty form of editing (refer to Figure 2) the following rules are performed:

- The frame indicating the identification of data scheme is generated.
- The file of documents (annotation) for data scheme is generated.
- For each specified data element in XSD the following is generated:
  - the frame indicating the element name and its location (in the XPath pattern) in the XML record structure;
  - The key button of hiding-revealing element in a form of editing;
  - The file of documents (annotation), if any, with an indication of a language;
  - The nested elements (for complex);
  - The field of entry element definition (for simple);
  - Names and data entry fields for each of potential attributes;
  - Key buttons for deleting (if allowed) or duplicating (if allowed) elements;
- The following key buttons are generated:
  - «Record»- for storage a result of editing;
  - «Clear»-for regeneration of empty editing form;
  - «Close»- for closing editing form without data storage.

The type of data and the placed restrictions are taken into account in the process of generation of data entry fields. In particular, the field of entry elements and attributes are presented with a list of dropdown definitions (refer to Figure 3) if there is XSD definitions such as:

Figure 2. Graphical interface of XML record editor

```
<xsd:simpleType name="recordTypeType">
    <xsd:restriction base="xsd:NMTOKEN">
        <xsd:enumeration value="Bibliographic"/>
        <xsd:enumeration value="Authority"/>
        <xsd:enumeration value="Holdings"/>
        <xsd:enumeration value="Classification"/>
        <xsd:enumeration value="Community"/>
    </xsd:restriction>
</xsd:simpleType>
```

If the XSD element contains indication for a pattern (RegEx), for example:

```
<xsd:simpleType name="indicatorDataType" id="ind.st">
    <xsd:restriction base="xsd:string">
        <xsd:whiteSpace value="preserve"/>
        <xsd:pattern value="[\da-z ]{1}"/>
    </xsd:restriction>
</xsd:simpleType>
```

In that case, the access to checking function of correspondence with a pattern of data entry in the form of editing is generated, that is XSLT code will be performed:

...
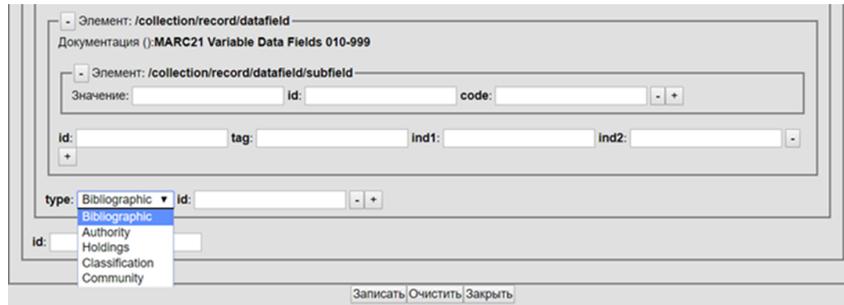
Figure 3. Graphical XML editor: data entry fields

```
<xsl:for-each select="xsd:simpleType/xsd:restriction/xsd:pattern">
    <xsl:attribute name="onChange">
            <xsl:text>e_change(this, /</xsl:text>
            <xsl:value-of select="@value"/>
            <xsl:text>/);</xsl:text>
    </xsl:attribute>
</xsl:for-each>
...
```

Which in turn generates the forms of elements

```
<input type="text"
            onChange="e_change(this, /[\da-z ]{1}/);" . . . />
```

A problem of recursive definitions arises from the described approach in XML formation on the ground of XSD. Recursivity may occur in the appliance of references to types and names. A fragment of a recursive determination is provided in the scheme with the help of the XSD.

```
\begin{flushleft}
<xsd:complexType name="organization">
   <xsd:sequence>
      <xsd:element name="id" type="int"/>
      <xsd:element name="name" type="string"/>
      <xsd:element name="sub-org" type="tns:organization"/>
   </xsd:sequence>
</xsd:complexType>
      <xsd:element name="region">
<xsd:complexType>
   <xsd:sequence>
      <xsd:element name="id" type="int" />
      <xsd:element ref="tns:region" />
   </xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
      <xsd:element name="record">
<xsd:complexType>
   <xsd:sequence>
      <xsd:element name="id" type="int" />
      <xsd:element name="organization" type="tns:organization" />
      <xsd:element ref="tns:region" />
   </xsd:sequence>
</xsd:complexType>
</xsd:element>
```

The XML elements with unrestricted length of Xpath are possible:

```
/record/organization/sub-org/sub-org/sub-org . . .

/record/region/region/region/region . . .
```

The attachment number control can be used for eliminating the endless number of item attachments in generation of graphic interfaces of editing records and for restricting them in accordance with the current demand. The list of processed elements, XSD (rules), is depicted by editor prototype in the table 1.

At the present time, the created prototype of mentioned the XML records adapting editor is being tested for various data charts. Furthermore, it is planned to boost its functional capabilities in the area of widening a list of supported elements of the XSD and JSON. At the end of the testing the editor will be built in the ZooSPACE-W subsystem of the ZooSPACE platform.

References

1. ANSI/NISO Z39.50-2003. Information Retrieval (Z39.50): Application Service Definition and Protocol Specification. - NISO Press, Bethesda, Maryland, U.S.A. — Nov 2002. — ISSN: 1041-5653. — ISBN: 1-880124-55-6.
2. SRU- Search/Retrieve via URL / The Library of Congress, 26.12.2013, http://www.loc.gov/standards/sru/
3. SRU - Explain Operation / The Library of Congress, 6.09.2013, http://www.loc.gov/standards/sru/explain/
4. Zhizhimov, O.L. : Explain Services on ZooSPACE Platform and Adaptive User Interfaces. CEUR Workshop Proceedings. - Vol.1536.,- ISSN 1613-0073. pp.30–36 - http://ceur-ws.org/Vol-1536/paper4.pdf.(2015)
5. ZeeRex: The Explainable "Explain"Service, http://zeerex.z3950.org
6. Zhizhimov, O.L., Fedotov, A.M., Shokhin, Y.I. The ZooSPACE platform- access organization to various distributed resources. Digital libraries: The Russian scientific e-magazine. - Vol.17. - № 2. - ISSN 1562-5419.(2014)
7. Zhizhimov, O.L., Lobykin, A.A., Turchanovskiy, I.Y., Panshin, A.A., Chudinov, S.A. Computer -assisted acquisition system of statistics event information in distributed information system. Vestnik NSU. Ser.: The Information technology. - Vol.11. - № 1. - pp.42–52. - ISSN 1818-7900.(2013)

Table 1. Supported XSD elements

Element	Attribute
annotation	
appinfo	
attribute	name, ref, type, use
choice	
complexContent	
complexType	name
documentation	
element	name, ref, type, substitutionGroup, maxOccurs, minOccurs
extension	base
group	name, ref, maxOccurs, minOccurs
import	namespace, schemaLocation
include	schemaLocation
list	itemType
restriction	base
schema	attributeFormDefault, elementFormDefault, blockDefault, finalDefault, targetNamespace, version, xmlns
sequence	maxOccurs, minOccurs
simpleContent	
simpleType	name
union	memberTypes
unique	

8. XML Schema Part 0: Primer Second Edition: W3C Recommendation 28 October 2004. `http://www.w3.org/TR/xmlschema-0/`

9. XML Schema Part 1: Structures Second Edition: W3C Recommendation 28 October 2004. `http://www.w3.org/TR/xmlschema-1/`

10. XML Schema Part 2: Datatypes Second Edition: W3C Recommendation 28 October 2004. `http://www.w3.org/TR/xmlschema-2/`

11. XSL Transformations (XSLT) Version 2.0: W3C Recommendation 23 January 2007. `http://www.w3.org/TR/xslt20/`

12. Mukhitova, A.A., Zhizhimov, O.L., Adaptive technologies in the context of designing administrative graphic interfaces for heterogeneous information systems of inputting and editing data // XVI Russian conference of 'The distributed informational-computational resources. Science in digital economy' (DICR-2017): Proceedings of XVI All-Russia Conference (4-7 of December, 2017) Novosibirsk / under the editorship of Zhizhimov O.L., Fedotov A.M.- Novosibirsk: ICT SB RAS.- pp.142–149.- ISBN: 978-5-905569-10-4.- http://elib.ict.nsc.ru/jspui/bitstream/ICT/1467/20/paper16.pdf(2017)

Filling up Link Grammar Parser dictionaries by using Word2vec techniques *

F. A. Murzin[1,2], M. J. Tussupova[1], A. S. Yerimbetova[1,3,4]

[1] Novosibirsk State University, 1, Pirogova str., Novosibirsk, 630090, Russia
[2] A.P. Ershov Institute of Informatics Systems SB RAS
6, Acad. Lavrentjev pr., Novosibirsk, 630090, Russia
[3] Institute of Information and Computational Technologies
125, Pushkin str. Almaty, 050010,Kazakhstan
[4] Kazakh academy of transport and communications named after M. Tynyshpayev
97, Shevchenko str., Almaty, 050012, Kazakhstan

Abstract The paper describes the methods of filling dictionaries of Link Grammar Parser with the help of machine learning. Vector model is used, which means the algebraic model for a representation of the text documents in the form of vectors. Vector model has been offered in 1975 by Gerard Salton. In this model, the document is considered as a set of terms, i.e. selected words or word-combinations. Every component of a vector of features corresponds to the separate term. The algorithm consists of two parts: definition of parts of speech of words and finding connectors between words in a sense of a link grammar. It is shown that the best predictor for Verbs and Pronouns is Random Forest Classifier, for Nouns and Adjectives is SVM, etc.

Keywords: Natural Language Processing, Machine Learning algorithms, Word Embedding, Vectorization

Introduction

In this paper, the methods of filling up dictionaries of Link Grammar Parser with the help of machine learning [1] are considered.

One of the main goals is to construct the algorithms that can estimate the document relevance on the basis of the text structure analysis. It is important that estimation will be based on the context of the search query and not limited only by keywords, their similarity or frequency.

The second problem is text clustering and classification. It plays an important role in many applications. For example information retrieval, web search, spam filtering, etc. It is possible to use the semantic-syntactical relations between

words constructed by the software system Link Grammar Parser to solve the problem [1], which is based on the so-called link grammar.

The results of our research are planned to be used in information retrieval systems. The application of the methods here considered to studies of Turkic languages is briefly described.

In a natural way, a necessity is appeared of filling of dictionaries of Link Grammar Parser. The algorithm should consist of two parts: definition of words' parts of speech and finding connectors between words in a sense of a link grammar. Now in the full volume, the first part is realised. For example, it is shown that the best predictor for Verbs and Pronouns is Random Forest Classifier, for Nouns and Adjectives is SVM, for Adverbs, Prepositions and Determiners is Logistic Regression.

The next step of our investigations will be a creation of part of speech tagging algorithms for some Turkic languages, such as Kazakh and Turkish, and a development of algorithms for finding connectors between words in these languages in a sense of a link grammar.

Text representation models

Vector model is the algebraic model for a representation of the text documents in the form of vectors. It has been offered in 1975 by Gerard Salton [1]. In the vector model, the document is considered as a set of terms, i.e. selected words or word-combinations. Every component of the feature vector corresponds to the separate term. The numerical value of a component is called a weight of the term, which characterises the importance of the term for a representation of the given document. If the term does not meet in the document its weight in this document is equal to zero.

Further, all terms, which meet in documents of a processed collection, may be ordered. If write out the weight of all terms to some document, including what is not entered in this document, the vector will be obtained, which will be the representation of the given document in the vector space. The dimension of this vector, as well as the dimension of the whole space, is equal to a number of various terms in all collection, and it is identical to all documents.

More formally, it is possible to present this statement in the form of the formula:

$$\vec{d_i} = (w_{i1}, w_{i2}, \ldots, w_{in}); \tag{1}$$

where $\vec{d_i}$ is a vector representation of i-th document, w_{ij} is a weight of j-th term in i-th document, n is a total number of various terms in all documents of a collection.

The vector model has a number of lacks:

1) interrelations between words are lost;

2) it is difficult to reveal a similarity of documents if documents have identical sense, but consist of different words;

3) complexity of a representation of the big documents.

To eliminate the specified lacks, the model of a representation of the text in a form of a graph is offered. Such representation effectively simulates communications between terms and reflects the information on the document structure. For text representation in a form of a graph, we use a syntactic parser of English, German, Russian, Arabian, Persian and other languages.

Link Grammar Parser is a syntactic analyser of the English language developed by Daniel Sleator and Davy Temperley in 1990th at the Carnegie Mellon University, USA. It is based on the so-called link grammar. Note that, in general, the underlying theory differs from the classical theory of syntax. Having received a sentence, the system attributes it with a syntactic structure which consists of a set of marked links connecting the pairs of words. The detailed description of the system can be found in [2, 3]. Link Grammar Parser includes approximately 60000 dictionary forms. It allows us to analyse a huge part of syntactic constructions, including numerous rare expressions and idioms. The parser work is stable; it can skip a part of a sentence it cannot understand and define some structure for the rest of the sentence. It is capable to process an unknown lexicon, and do reasonable assumptions about the syntactic category of unknown words based on the context and writing. The parser contains data about various names, numerical expressions, and punctuation marks.

The rules of words connection are described in the set of dictionaries. For each word in a dictionary, it is fixed what are its connectors with other words in a sentence. A connector has a name with which the considered unit (word) can enter a sentence. For example, the mark S corresponds to communication between a subject and a predicate, O is a connector between an object and a predicate. There are more than one hundred most important basic connectors. To denote the direction of a connector, the sign "+" is used to indicate a right connector and the sign "-" to indicate a left connector.

In this article, we suggest filling dictionaries of Link Grammar Parser with the help of machine learning.

Text Preprocessing

The first stage is the text preprocessing, i.e. the preparation of the text for a subsequent analysis. Thereby several tasks are fulfilled:

1. Dividing the text into words/sentences;

2. Using a special software, namely, nltk library for a definition of parts of speech of words;

We choose Hamlet by W. Shakespeare as the text for testing algorithms. First, we need to split sentences into words for the next determination of its part of speech.

It is possible to do immediately during the article loading. Further, we pass to the lowercase, totally in the whole text, to avoid a difference between 'hamlet' and 'Hamlet', for example. Also, we remove the commonly used words.

	Words
0	tragedie
1	of
2	hamlet
3	actus
4	primus

Figure 1. Splitting the text into words.

The next step is to tag words. In nltk library, there are various types of parts of speech [4]. Some of them are not classical. The most important ones are given below.

1. CC | Coordinating conjunction |
2. CD | Cardinal number |
3. DT | Determiner |
4. IN | Preposition or subordinating conjunction |
5. JJ | Adjective |
6. JJR | Adjective, comparative |
7. JJS | Adjective, superlative |
8. MD | Modal |
9. NN | Noun, singular or mass |
10. NNS | Noun, plural |
11. PRP | Personal pronoun |
12. PRP$ | Possessive pronoun |
13. RB | Adverb |
14. RBR | Adverb, comparative |
15. TO | to |
16. VB | Verb, base form |
17. VBD | Verb, past tense |
18. VBG | Verb, gerund or present participle |
19. VBN | Verb, past participle |
20. VBP | Verb, non-3rd person singular present |
21. VBZ | Verb, 3rd person singular present |
22. WDT | Wh-determiner |
23. WP | Wh-pronoun |
24. WP$ | Possessive wh-pronoun |
25. WRB | Wh-adverb |

We take into account only the main attributes from this list: noun, pronoun, adjective, verb, adverb, preposition, determiner, numerical. As a result, we obtain the table represented below.

	Verb	Noun	Pronoun	Adjective	Adverb	Preposition	Determiner
0	0	1	0	0	0	0	0
1	0	0	0	0	0	1	0
2	0	1	0	0	0	0	0
3	0	1	0	0	0	0	0
4	0	1	0	0	0	0	0

Figure 2. Table of words parts of speech. Compressed Version.

In our case, there is no any column marked by "Numeric". The reason is that there is no numeric information in our considered text.

Word Embedding

Word embedding is the collective name for a set of language modelling and feature learning techniques in Natural Language Processing [5]. Using this kind of techniques we can consider a vector model of word, sentence, text. There are several corresponding program tools.

Word2vec could be a gather of related models that are utilised to deliver word embeddings. Often these models use two-layer neural systems. Word2vec takes as its input a huge corpus of content and usually produces a vector space having a dimension of several hundred. For each interesting word within the corpus being, some vector in the space is assigned. Word vectors are situated within the vector space such that words that share common settings within the corpus are found near to one another in the vector space. Namely, such situation is described in section 2. More exact, there are two types of word embeddings: vector and graph. We consider the vector representation.

Given a text corpus, the word2vec tool learns a vector for every word in the vocabulary using the Continuous Bag-of-Words or the Skip-Gram neural network architectures. The user should to specify the following:

– desired vector dimensionality
– the size of the context window for either the Skip-Gram or the Continuous Bag-of-Words model
– training algorithm: hierarchical softmax and / or negative sampling
– threshold for down sampling the frequent words
– number of threads to use
– the format of the output word vector file (text or binary)

Frequency-inverse report recurrence $(tf - idf)$ could be a numerical measurement that's aiming to reflect how critical a word is to a record in a collection or corpus. It is frequently utilized as a weighting figure in looks of data recovery, content mining, and client modelling. The $tf-idf$ esteem increases proportionally to the number of times a word shows up within the archive and is balanced by the recurrence of the word within the corpus, which helps to alter for the reality that a few words show up more habitually in common. $Tf-idf$ is one of the foremost prevalent term-weighting plans nowadays; 83% of text-based recommended frameworks in advanced libraries utilize $tf - idf$.

In a large text corpus, some words will be very often present, e.g. articles, prepositions, auxiliary verbs/words as "the "a "is"in English. These words are carrying very little meaningful information about the actual contents of the document. We should omit them, otherwise, those very frequent terms could be "shadows"of the frequencies of rarer yet more interesting terms.

In order to re-weight the numerical floating point values of features suitable for application of a classifier, it is very common the use of $tf - idf$ transform.

Tf means a term-frequency in the document, while $tf - idf$ means a term-inverse document-frequency, that means the inversion of a frequency with which some word meets in the collection of documents:

$$tf - idf(t, d) = tf(t, d) * idf(t); \qquad (2)$$

Experiments

In the experiments, Word2Vec vectorizer was used with several machine learning classifiers such as Logistic Regression, Decision Tree Classifier, SVM, Random Forest Classifier and KNeighborsClassifier. Each of them we have scored with ROC AUC. Shortly these methods are described in [6].

Results for Logistic Regression:
ROC AUC score for Verb = 0.8559714533139516
ROC AUC score for Noun = 0.6777486340249596
ROC AUC score for Pronoun = 0.9072944804503941
ROC AUC score for Adjective = 0.6542860370501943
ROC AUC score for Adverb = 0.8388773824383644
ROC AUC score for Preposition = 0.7729856769704812
ROC AUC score for Determiner = 0.7641189437754812
Results for Decision Tree Classifier:
ROC AUC score for Verb = 0.6548053555964495
ROC AUC score for Noun = 0.5910786095957337
ROC AUC score for Pronoun = 0.531827687160184
ROC AUC score for Adjective = 0.5594783169178744
ROC AUC score for Adverb = 0.6478730201325129
ROC AUC score for Preposition = 0.5126563189876151
ROC AUC score for Determiner = 0.49648162194620904
Results for SVM:

ROC AUC score for Verb = 0.8601630429179682
ROC AUC score for Noun = 0.7098495939578038
ROC AUC score for Pronoun = 0.8221093302054803
ROC AUC score for Adjective = 0.6962292892336787
ROC AUC score for Adverb = 0.8031237254181034
ROC AUC score for Preposition = 0.5748435959300228
ROC AUC score for Determiner = 0.6475326317120192
Results for Random Forest Classifier:
ROC AUC score for Verb = 0.8315995461293415
ROC AUC score for Noun = 0.6450500860244069
ROC AUC score for Pronoun = 0.865155045707116
ROC AUC score for Adjective = 0.6231781980909588
ROC AUC score for Adverb = 0.7879387790229063
ROC AUC score for Preposition = 0.6414009903738194
ROC AUC score for Determiner = 0.6832767216586074
KNN results:
ROC AUC score for Verb = 0.7046526993834988
ROC AUC score for Noun = 0.6317347848624765
ROC AUC score for Pronoun = 0.5295712619063342
ROC AUC score for Adjective = 0.6206376313253099
ROC AUC score for Adverb = 0.685744948284865
ROC AUC score for Preposition = 0.5241368375468388
ROC AUC score for Determiner = 0.5589365459372333

We can see that the best predictor for Verbs and Pronouns is Random Forest Classifier, for Nouns and Adjectives is SVM, for Adverbs, Prepositions and Determiners is Logistic Regression.

Conclusion and future work

Algorithms of Machine Learning and Vectorization are working successfully. They may be used to fill up various dictionaries such as dictionaries of Link Grammar Parser. The best predictor for Verbs and Pronouns is Random Forest Classifier, for Nouns and Adjectives is SVM, for Adverbs, Prepositions and Determiners is Logistic Regression. There is no need to use KNN - unsupervised learning and Decision Tree Classifier.

The next step of our investigations is the creation of part of speech tagging algorithms for some Turkic languages: Kazakh, Turkish, etc. and finding connectors between words in these languages with respect to a link grammar. Our work is based on the results from articles [7,8].

References

1. Salton, G., Yang, K.S., Yu, C.T.: A theory of term inportance in automatic text analysis. Journal of the ASIS. Vol. 26 , No. 1, - pp. 33-44.(1975)

2. Sleator, D., Temperley, D.: Parsing English with a Link Grammar. Carnegie Mellon University Computer Science technical report CMU-CS-91-196, October. - 93 p.(1991)
3. Temperley, D., Sleator, D., Lafferty, J.: Link Grammar Documentation. `http://www.link.cs.cmu.edu/link/dict/index.html(accessed:15November2017)` (1998)
4. Ide, N., Veronis, J.: Introduction to the special issue on word sense disambiguation: the state of the art. Computational Linguistics. Vol. 24, No. 1, - pp. 2-40.(1998)
5. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed Representations of Words and Phrases and their Compositionality. `http://scikit-learn.org/stable/modules/feature_extraction.html` - 9 p. (2013)
6. Levchenko, S.V.: Development of the words clusterization method under semantic characteristics with using Word2vec algorithms. New information technologies in the automated systems. - No. 20. - 5 p. (2017)
7. Batura, T.V., Murzin, F.A., Semich, D.F., Sagnayeva, S.K., Tazhibayeva, S.Zh., Bakiyev, M.N., Yerimbetova, A.S., Bakiyeva, A.M.: Using the Link grammar parser in the study of Turkic languages. Eurasian journal of mathematical and computer applications. Astana: L.N. Gumilyov Eurasian National University, Vol. 4. Iss. 2. pp. 14-22. (2016)
8. Batura, T.V., Murzin, F.A., Sagnayeva, S.K., Tazhibayeva, S.Zh., Yerimbetova, A.S., Bakiyeva, A.M.: Link Grammar Parser for Turkic Languages and algorithms for estimation the relevance of documents. 2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT-2016). 12-14 October 2016, Baku, Azerbaijan. pp. 104-107. (2016).

Dynamical effects of a magnetic field versus the nonlinear wave regime of a rotating layer of liquid

S. I. Peregudin[1], E. S. Peregudina[2], and S. E. Kholodova[3]

[1] St. Petersburg State University, Saint Petersburg, Russia
[2] St. Petersburg Mining University, Saint Petersburg, Russia
[3] ITMO University, Saint Petersburg, Russia
peregudinsi@yandex.ru,ehllina-peregudina@yandex.ru,kholodovase@yandex.ru

Abstract The present paper is concerned with the dynamics of a magnetic field consequent on three-dimensional large-scale motions of an inviscid incompressible homogeneous perfectly conducting rotating fluid concentrated in a spherical layer. The proposed mathematical model of the above physical process is a closed system of partial differential equations consisting of hydrodynamic equations with due account of the rotation, the Lorentz force, and the corresponding equations of magnetic dynamics with required boundary conditions. We analyze the mathematical model which can be used for calculation of three-dimensional motions with large time scale and when the space horizontal scale is comparable to the layer radius. The principal idea of our approach is in the construction of a scheme of successive approximation, in which the geostrophic approximation is the first step. Our approach allows one to go beyond the heuristic arguments and derive general geostrophic equations describing the motion of both homogeneous and inhomogeneous electrically conducting rotating fluid. We obtain an analytic solution of the system of nonlinear partial differential equations that model the geostrophic motion in the spherical layer of perfect electrically conducting inhomogeneous rotating fluid. The analysis of the structure of the above fields of magnetohydrodynamic quantities allows one to justify the existence of strong changes in the thing layer adjacent to the outer boundary.

Keywords: quasi-geostrophic motion, rotating fluid, magnetic field diffusion, magnetohydrodynamic processes.

Introduction

This paper is concerned with the study of wave three-dimensional large-scale motions of an inviscid incompressible inhomogeneous perfectly conducting rotating fluid concentrated in a spherical domain. The results of our study can be used in astrophysics and geophysics, and in particular, in the analysis of the processes occurring in the Earth liquid core and in stellar interiors. Our findigns

may prove useful not only in geophysical applications, but may also be important in the study of the self-excitation process from a magnetohydrodynamic dynamo in relatively large masses of liquid metal and in engineering devices, for example, in engineering processes involving pressure chambers of fast-neutron reactors, blast furnaces, titanium production reactors, etc.

Problems of generation of magnetic fields frequently involve processes pertaining to the form of the motion and conditions under which magnetic fields can exist for sufficiently large time intervals; i.e., we are interested in the character of the processes under consideration.

In the studies in these field, by a large time one means a time considerably smaller than the magnetic field decay time, but at the same time it should be sufficiently large in order that some generations would be able to observe this decay.

Of course, the above estimates should be properly qualified. It may well be that in the actual fact they have even larger orders, because as of now magnetic field decay was never vividly manifested, although some decreases of its intensity were observed, but each decrease was always followed by an increase in the intensity. Hence, even if the statement about the decay of the Earth magnetic field is applicable to the Earth conditions and is possible, then most likely the order of the time under consideration is much larger than than those from the available sources or nonetheless all the processes of interest for us represent steady-state oscillations.

The paper [1] was concerned with the role of stratification in the principal dynamics of the magnetohydrodynamic processes under consideration. The analysis of the obtained solution enabled one to prove the existence of a steady-state of oscillations for large values of time, which justifies the important role of stratification of density of the liquid layer which controls in a number of cases its principal dynamics.

The magnetic field dynamics depends substantially on the motion of the fluid under consideration directly in the thin layer adjacent to the boundary. Earlier an analysis was carried out and the conclusion was made on the onset of perturbations of magnetohydrodynamic fields with relatively small horizontal scale which is considerably smaller than the radius of the layer developing near some point with spherical coordinates (θ_0, φ_0). Of special is the solution of the similar problem in a layer whose horizontal scale is of the order of the layer radius. Such problems will be considered in the present study.

Thus, we shall consider wave motions near the outer boundary of a thin spherical layer of an inhomogeneous fluid.

Principal Equations of Large-scale Geostrophic Motions of an Electrically Conducting Fluid

The motion of an inviscid perfectly conducting incompressible inhomogeneous fluid in a system rotating with angular velocity ω is described in magnetic

hydrodynamics in Euler variables by the equations (see [1]–[3])

$$\frac{d\rho}{dt} + \rho \operatorname{div} \mathbf{v} = 0, \qquad \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{\nabla p}{\rho} - 2\boldsymbol{\omega} \times \mathbf{v} - g\mathbf{z} + \frac{1}{\mu \rho} \operatorname{rot} \mathbf{b} \times \mathbf{b},$$

$$\frac{\partial \mathbf{b}}{\partial t} = \operatorname{rot} (\mathbf{v} \times \mathbf{b}), \qquad \operatorname{div} \mathbf{b} = 0,$$

where $\mathbf{b}$ is the vector of magnetic induction vector, $\mathbf{v}$ is the fluid velocity in the frame rotating with angular velocity $\boldsymbol{\omega}$, p is the pressure, ρ is the density, g is the acceleration of gravity.

It is assumed that the magnetic permeability μ is constant,

$$R_m = \frac{LU}{\widetilde{\lambda}} \gg 1,$$

where L, U are the characteristic size and velocity, $\widetilde{\lambda}$ is the magnetic diffusion coefficient. The case $R_m \gg 1$ is realized in the Earth liquid core; it also frequently occurs in astrophysics and in some engineering processes.

If the density is considered variable, then in addition to the motion equation, the continuity equation and Maxwell equations, one needs to invoke the balance equation for the internal energy [2]–[5]

$$\rho \frac{d\mathcal{E}}{dt} = -p\rho \frac{d}{dt} \left(\frac{1}{\rho} \right) + k\Delta T + \chi + \rho Q + \widetilde{\lambda} (\operatorname{rot} \mathbf{b})^2 \qquad (1)$$

and the state equation

$$\rho = \rho_0 \left(1 - \alpha(T - T_0) \right),$$

where ρ_0 and T_0 are, respectively, the mean density of fluid and the mean temperature, $\alpha = -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial T} \right) \bigg|_p$ is the thermal-expansion coefficient. For an incompressible fluid, instead of equation (1) we shall use the heat-transfer equation in the form [4]

$$\frac{dT}{dt} = \frac{k}{\rho c_P} \Delta T + \frac{Q}{c_P}; \qquad (2)$$

here c_P is the specific heat capacity with constant pressure. Then the temperature in equation (2) can be expressed in terms of the density

$$\frac{d\rho}{dt} = \varkappa \Delta \rho - \frac{\alpha \rho_0}{c_P} Q,$$

here $\varkappa = \dfrac{k}{\rho c_P}$ is the thermometric conductivity.

Consider the dimensionless variables

$$r_* = r_0 \left(1 + \frac{D}{r_0} z \right), \qquad v_{\lambda_*} = U v_\lambda, \qquad v_{\theta_*} = U v_\theta, \qquad b_{\lambda_*} = B b_\lambda,$$

$$b_{\theta_*} = B b_\theta, \quad v_{r_*} = \frac{D}{r_0}, \quad U v_r = W v_r, \quad b_{r_*} = \frac{D}{r_0} B b_r = V b_r, \quad t_* = \frac{r_0}{U} t,$$

here r_0 is the radius of the spherical layer, D is the depth scale of the motion. As the horizontal sale of the motion, we shall use the radius r_0, U is the horizontal velocity scale, $W = \dfrac{D}{r_0} U$ is the vertical velocity scale, B is the scale of horizontal components of the magnetic field, $V = \dfrac{D}{r_0} B$ is the scale of the vertical component of the magnetic field.

We use the spherical coordinates r, θ, λ, where

$$r \geq 0, \quad 0 \leq \theta \leq \pi, \quad 0 \leq \lambda \leq 2\pi.$$

In the dimensionless coordinates, the motion equations assume the form

$$\varepsilon F \frac{d\rho}{dt} + (1 + \varepsilon F \rho) \left[\frac{\partial v_z}{\partial z} + \frac{2 D v_z}{r_*} + \frac{r_0}{r_* \sin\theta} \left(\frac{\partial}{\partial \theta} (v_\theta \sin\theta) + \frac{\partial v_\lambda}{\partial \lambda} \right) \right] = 0,$$

$$\varepsilon \left(\frac{dv_\lambda}{dt} + v_\lambda v_z \frac{D}{r_*} + v_\lambda v_\theta \frac{r_0}{r_*} \cot\theta \right) + v_\theta \cos\theta + \frac{D}{r_0} v_z \sin\theta =$$

$$= -\frac{r_0}{r_* \sin\theta} \frac{1}{1 + \varepsilon F \rho} \left[\frac{\partial p}{\partial \lambda} + \frac{\varepsilon B^2}{U^2 \mu \rho_0} \left(b_\theta \frac{\partial b_\theta}{\partial \lambda} + \left(\frac{D}{r_0} \right)^2 b_z \frac{\partial b_z}{\partial \lambda} \right) \right] +$$

$$+ \frac{\varepsilon B^2}{U^2 \mu \rho_*} \left(b_z \frac{\partial b_\lambda}{\partial z} + \frac{r_0}{r_*} b_\theta \frac{\partial b_\lambda}{\partial \theta} + \frac{D}{r_0} \frac{r_0}{r_*} b_\lambda b_z + \frac{r_0}{r_*} b_\lambda b_\theta \cot\theta \right),$$

$$\varepsilon \left(\frac{dv_\theta}{dt} + v_\theta v_z \frac{D}{r_*} - v_\lambda{}^2 \frac{r_0}{r_*} \cot\theta \right) - v_\lambda \cos\theta = -\frac{r_0}{r_*} \frac{1}{1 + \varepsilon F \rho} \left[\frac{\partial p}{\partial \theta} + \frac{\varepsilon B^2}{U^2 \mu \rho_0} \times \right.$$

$$\times \left. \left(b_\lambda \frac{\partial b_\lambda}{\partial \theta} + \left(\frac{D}{r_0} \right)^2 b_z \frac{\partial b_z}{\partial \theta} - \frac{D}{r_0} b_\theta b_z + \cot\theta \, b_\lambda^2 \right) \right] + \frac{\varepsilon B^2}{U^2 \mu \rho_*} \left(b_z \frac{\partial b_\theta}{\partial z} + \frac{r_0}{r_* \sin\theta} b_\lambda \frac{\partial b_\theta}{\partial \theta} \right),$$

$$(1 + \varepsilon F \rho) \left\{ \varepsilon \left(\frac{D}{r_0} \right)^2 \frac{dv_z}{dt} - \varepsilon \frac{D}{r_*} \left(v_\lambda{}^2 + v_\theta{}^2 \right) - \frac{D}{r_0} v_\lambda \sin\theta \right\} = -\frac{\partial p}{\partial z} - \rho -$$

$$- \frac{\varepsilon B^2}{U^2 \mu \rho_0} \left(b_\lambda \frac{\partial b_\lambda}{\partial z} + b_\theta \frac{\partial b_\theta}{\partial z} \right) + \frac{\varepsilon B^2}{U^2 \mu \rho_0} \left(\frac{D}{r_0} \right)^2 \frac{r_0}{r_*} \left[b_\theta \frac{\partial b_z}{\partial \theta} + \frac{1}{\sin\theta} \frac{\partial b_z}{\partial \lambda} - \frac{r_0}{D} \left(b_\lambda{}^2 + b_\theta{}^2 \right) \right],$$

$$\frac{d\rho}{dt} = \frac{k_V r_0}{U D^2} \frac{\partial^2 \rho}{\partial z^2} + \frac{k_H}{U r_0} \Delta_2 \rho,$$

$$\frac{\partial b_z}{\partial t} = \frac{r_0}{r_*} b_\theta \frac{\partial v_z}{\partial \theta} + \frac{r_0}{r_* \sin\theta} b_\lambda \frac{\partial v_z}{\partial \lambda} + b_z \frac{\partial v_z}{\partial z} - \frac{r_0}{r_*} v_\theta \frac{\partial b_z}{\partial \theta} - \frac{r_0}{r_* \sin\theta} v_\lambda \frac{\partial b_z}{\partial \lambda} - v_z \frac{\partial b_z}{\partial z},$$

$$\frac{\partial b_\theta}{\partial t} = \frac{r_0}{r_*} b_\theta \frac{\partial v_\theta}{\partial \theta} + \frac{r_0}{r_* \sin\theta} b_\lambda \frac{\partial v_\theta}{\partial \lambda} + b_z \frac{\partial v_\theta}{\partial z} - \frac{r_0}{r_*} v_\theta \frac{\partial b_\theta}{\partial \theta} - \frac{r_0}{r_* \sin\theta} v_\lambda \frac{\partial b_\theta}{\partial \lambda} - v_z \frac{\partial b_\theta}{\partial z},$$

$$\frac{\partial b_\lambda}{\partial t} = \frac{r_0}{r_*} b_\theta \frac{\partial v_\lambda}{\partial \theta} + \frac{r_0}{r_* \sin\theta} b_\lambda \frac{\partial v_\lambda}{\partial \lambda} + b_z \frac{\partial v_\lambda}{\partial z} - \frac{r_0}{r_*} v_\theta \frac{\partial v_\lambda}{\partial \theta} - \frac{r_0}{r_* \sin\theta} v_\lambda \frac{\partial b_\lambda}{\partial \lambda} - v_z \frac{\partial b_\lambda}{\partial z},$$

where

$$\varepsilon = \frac{U}{2\,\omega r_0}, \qquad F = \frac{4\omega^2 {r_0}^2}{gD},$$

k_V and k_H are, respectively, the diffusion coefficients in the vertical and horizontal directions. It is assumed that k_V and k_H are distinct. The operators $\dfrac{d}{dt}$ and Δ_2 are defined as

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \frac{r_0}{r_*}\left(\frac{v_\lambda}{\sin\theta}\frac{\partial}{\partial\lambda} + v_\theta\frac{\partial}{\partial\theta}\right) + v_z\frac{\partial}{\partial z},$$

$$\Delta_2 = \frac{{r_0}^2}{{r_*}^2}\left(\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial}{\partial\theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\lambda^2}\right).$$

For $U = 1\,\mathrm{cm/s}$, $r_0 = 6{\cdot}10^8\,\mathrm{cm}$, $D = 40\,\mathrm{km}$, $2\omega \approx 1,4{\cdot}10^{-4}\,\mathrm{s}^{-1}$, the parameters ε, F and $\dfrac{D}{r_0}$ are, respectively, as follows:

$$\varepsilon = O(10^{-5}), \quad F = 1,8, \quad \frac{D}{r_0} = O(10^{-3}).$$

Making

$$M_A = \frac{U\sqrt{\mu\rho}}{B} = O(1),$$

and representing the sought-for functions as a power series with respect to the small Rossby number for the principal terms of the corresponding expansions, the motion equations read as

$$v_\theta\cos\theta = -\frac{1}{\sin\theta}\frac{\partial p}{\partial\lambda}, \qquad v_\lambda\cos\theta = \frac{\partial p}{\partial\theta}, \qquad \rho = -\frac{\partial p}{\partial z},$$

$$\frac{\partial v_z}{\partial z} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(v_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial v_\lambda}{\partial\lambda} = 0, \qquad \frac{\partial\rho}{\partial t} + \frac{v_\lambda}{\sin\theta}\frac{\partial\rho}{\partial\lambda} + v_\theta\frac{\partial\rho}{\partial\theta} + v_z\frac{\partial\rho}{\partial z} = \nu\frac{\partial^2\rho}{\partial z^2},$$

$$\frac{\partial b_z}{\partial z} + \frac{\partial b_\theta}{\partial\theta} + b_\theta\cot\theta + \frac{1}{\sin\theta}\frac{\partial b_\lambda}{\partial\lambda} = 0,$$

$$\frac{\partial b_z}{\partial t} = b_\theta\frac{\partial v_z}{\partial\theta} + \frac{b_\lambda}{\sin\theta}\frac{\partial v_z}{\partial\lambda} + b_z\frac{\partial v_z}{\partial z} - v_\theta\frac{\partial b_z}{\partial\theta} - \frac{v_\lambda}{\sin\theta}\frac{\partial b_z}{\partial\lambda} - v_z\frac{\partial b_z}{\partial z},$$

$$\frac{\partial b_\theta}{\partial t} = b_\theta\frac{\partial v_\theta}{\partial\theta} + \frac{b_\lambda}{\sin\theta}\frac{\partial v_\theta}{\partial\lambda} + b_z\frac{\partial v_\theta}{\partial z} - v_\theta\frac{\partial b_\theta}{\partial\theta} - \frac{v_\lambda}{\sin\theta}\frac{\partial b_\theta}{\partial\lambda} - v_z\frac{\partial b_\theta}{\partial z},$$

$$\frac{\partial b_\lambda}{\partial t} = b_\theta\frac{\partial v_\lambda}{\partial\theta} + \frac{b_\lambda}{\sin\theta}\frac{\partial v_\lambda}{\partial\lambda} + b_z\frac{\partial v_\lambda}{\partial z} - v_\theta\frac{\partial b_\lambda}{\partial\theta} - \frac{v_\lambda}{\sin\theta}\frac{\partial b_\lambda}{\partial\lambda} - v_z\frac{\partial b_\lambda}{\partial z}.$$

So, the problem is to find the density distribution and all magnetohydrodynamic quantities appearing as a result of thermodynamic changes near the domain boundary under the assumption that such a problem is described by the equations of large-scale dynamics that were obtained earlier.

Construction of Nonstationary Solution with due Regard of Dissipation

We shall construct spiral perturbations of the system for which $\dfrac{\partial}{\partial\lambda} = 0$, $b_\lambda = 0$. Then the original system assumes the form

$$v_\theta = 0, \quad v_\lambda = \frac{1}{\cos\theta}\frac{\partial p}{\partial\theta}, \quad \frac{\partial v_z}{\partial z} = 0, \quad \rho = -\frac{\partial p}{\partial z}, \quad v_z\frac{\partial^2 p}{\partial z^2} = \nu\frac{\partial^3 p}{\partial z^3} - \frac{\partial^2 p}{\partial t\partial z},$$

$$\frac{\partial b_z}{\partial t} = b_\theta\frac{\partial v_z}{\partial\theta} - v_z\frac{\partial b_z}{\partial z}, \quad \frac{\partial b_\theta}{\partial t} = -v_z\frac{\partial b_\theta}{\partial z}, \quad b_\theta\frac{\partial v_\lambda}{\partial\theta} + b_z\frac{\partial v_\lambda}{\partial z} = 0,$$

$$\frac{\partial b_z}{\partial z} + \frac{\partial b_\theta}{\partial\theta} + b_\theta\cot\theta = 0$$

with the boundary conditions

$$v_\lambda \to 0 \qquad\qquad z \to -\infty,$$
$$b_z = b_z^{(e)}(\theta, t) \qquad\qquad z = 0,$$
$$b_\theta = b_\theta^{(e)}(\theta, t) \qquad\qquad z = 0.$$

Integrating this system, we get the following representations of the magnetohydrodynamic characteristics

$$v_\theta = 0, \quad v_\lambda(\theta, z, t) = -\alpha\sin\theta\exp\left(z - \int v_z(\theta, t)dt\right), \quad v_z(\theta, t) = v_{ze}(\theta, t),$$

$$p(\theta, z, t) = -\frac{\alpha\exp(z)}{2}\int\sin 2\theta\exp\left(-\int v_z(\theta, t)dt\right)d\theta + \exp(z)C_2(t),$$

$$\rho(\theta, z, t) = \frac{\alpha}{2}\exp(z)\int\sin 2\theta\exp\left(-\int v_z(\theta, t)dt\right)d\theta - \exp(z)C_2(t),$$

$$b_\lambda(\theta, z, t) = 0, \qquad b_\theta(\theta, z, t) = \exp\left(z - \int v_z(\theta, t)dt\right),$$

$$b_z(\theta, z, t) = \exp\left(z - \int v_z(\theta, t)dt\right)\left[\int\frac{\partial v_z}{\partial\theta}dt - \cot\theta\right] + \widetilde{C}_1(\theta),$$

where the function $C_2(t)$ is determined from the relation

$$v_z(\theta, t) = \frac{\nu(T_s - C_2) + C_2 - \dfrac{\alpha}{2}\displaystyle\int\sin 2\theta v_z\exp\left(-\int v_z(\theta, t)dt\right)d\theta - C_2'(t)}{T_s(\theta, t)}.$$

In particular, for $C_2(t) = 0$

$$v_z(\theta, t) = \nu - \frac{\alpha}{2T_s}\int v_z\sin 2\theta\exp\left(-\int v_z(\theta, t)dt\right)d\theta$$

and so

$$T_s(\theta, t) = \frac{\alpha\int v_z\sin 2\theta\exp\left(-\int v_z(\theta, t)dt\right)d\theta}{2(\nu - v_z)}.$$

References

1. Kholodova, S.E.: Quasi-geostrophic motions in a rotating layer of an electrically conducting fluid. Journal of Applied Mechanics and Technical Physics 50, no. 1., pp. 25–34 (2009).
2. Kholodova, S.E., Peregudin, S.I.: Modeling and Analysis of Streams and Waves in Liquid and Granular Mediums. St. Peterb. Gos. Univ., St. Petersburg (2009) [in Russian].
3. Needler, G.T.: A model for thermocline circulation in an ocean of finite depth. J. Marine Res. 25, pp. 329–342 (1967)
4. Landau, L. D., Lifshitz, E. M., Pitaevskii, L.P.: Electrodynamics of Continuous Media. Vol. 8 (2nd ed.). Butterworth-Heinemann (1984)
5. Pedlosky J. Geophysical Fluid Dynamics. New York, Springer (1987)

On the stability of wave processes
in a rotating electrically conducting fluid

S. I. Peregudin[1], E. S. Peregudina[2], and S. E. Kholodova[3]

[1] St. Petersburg State University, Saint Petersburg, Russia
[2] St. Petersburg Mining University, Saint Petersburg, Russia
[3] ITMO University, Saint Petersburg, Russia
peregudinsi@yandex.ru,ehllina-peregudina@yandex.ru,kholodovase@yandex.ru

Abstract The purpose of the present study is to reduce a system of
partial differential equations that model perturbations in a layer of
perfect electrically conducting rotating fluid bounded by space- and
time-varying surfaces with due account of the magnetic field diffusion
and inertia forces. For the reduced equations we construct solutions
describing the propagation of small-amplitude waves in a horizontal
infinitely extended layer and in a long narrow channel.

Keywords: quasi-geostrophic motion, rotating fluid, motion stability,
large-scale wave processes, reduction of a system of equations, magneto-
hydrodynamic processes.

Introduction

Large-scale motion of electrically conducting fluid was considered in a number
of studies concerned with the model constructed in the fast rotation limit.
In the framework of this theory, the inertial force is neglected in the motion
equation. As a result, inertia waves, Alfvén waves, and Rossby waves are
rejected. Moreover, in the fast rotation limit, the velocity is determined not
uniquely, but up to an additive term representing the geostrophic velocity. The
last fact stems from the observation that the geostrophic velocity does not
satisfy the magnetostrophic equation. This difficulty can be circumvented by an
artificial introduction of viscous forces and by disregarding the viscosity in cases
when it is possible. Earlier the problem of large-scale motion of an electrically
conducting fluid was studied in the layer between layers in the magnetostrophic
approximation with due account of the viscous forces. In the present study, it
is assumed that the layer boundaries are not planar, but are rather given by
space- and time-varying surfaces. Additionally, we solve the complete system
of magnetohydrodynamic equations taking into account the inertia forces in the
motion equations and assuming that in the magnetic field equations the diffusive
terms have the same order as the convection ones — in other words, the problem
under consideration is solved with an arbitrary value of the magnetic Reynolds
number. By introduction of auxiliary functions, it proves possible to reduce the
system of partial differential equations to one scalar equation, for which an exact

solutions is given. From this solution one can draw a conclusion on the influence of the magnetic field diffusion effects on its regeneration and preservation in the absence of an external field.

The results of this study can be used in astrophysics and geophysics, and in particular, in the study of processes taking place in the Earth liquid core and stellar interiors, and well as in the consideration of self-exciting processes of magnetohydrodynamic dynamo in relatively large masses of liquid metal and in engineering devices — for example, in engineering processes involving pressure chambers of fast-neutron reactors.

Fundamental Horizontal Structure Equations

Consider a thin layer of electrically conducting incompressible fluid which rotates with angular velocity $\boldsymbol{\omega}$ and which is bounded from below by a moving bed referenced to the level $z = 0$ by the surface $z = -h_B(x,y,t)$ with unknown function $h_B(x,y,t)$, and which is bounded from above by a known surface $z = -Z(x,y)$. The rotation axis agrees with the z-axis; i.e., $\boldsymbol{\omega} = \omega\mathbf{k}$. In the projections to the coordinate axes, the fundamental equations of magnetic hydrodynamics of the problem under consideration read as [1]

$$\frac{\partial v_x}{\partial t} + v_x\frac{\partial v_x}{\partial x} + v_y\frac{\partial v_x}{\partial y} + v_z\frac{\partial v_x}{\partial z} = -\frac{1}{\rho}\frac{\partial}{\partial x}\left(p + \frac{b^2}{2\mu}\right) +$$
$$+ 2\omega v_y + \frac{1}{\mu\rho}\left(b_x\frac{\partial b_x}{\partial x} + b_y\frac{\partial b_x}{\partial y} + b_z\frac{\partial b_x}{\partial z}\right), \tag{1}$$

$$\frac{\partial v_y}{\partial t} + v_x\frac{\partial v_y}{\partial x} + v_y\frac{\partial v_y}{\partial y} + v_z\frac{\partial v_y}{\partial z} = -\frac{1}{\rho}\frac{\partial}{\partial y}\left(p + \frac{b^2}{2\mu}\right) -$$
$$- 2\omega v_x + \frac{1}{\mu\rho}\left(b_x\frac{\partial b_y}{\partial x} + b_y\frac{\partial b_y}{\partial y} + b_z\frac{\partial b_y}{\partial z}\right), \tag{2}$$

$$\frac{\partial v_z}{\partial t} + v_x\frac{\partial v_z}{\partial x} + v_y\frac{\partial v_z}{\partial y} + v_z\frac{\partial v_z}{\partial z} = -\frac{1}{\rho}\frac{\partial}{\partial y}\left(p + \frac{b^2}{2\mu}\right) - g +$$
$$+ \frac{1}{\mu\rho}\left(b_x\frac{\partial b_z}{\partial x} + b_y\frac{\partial b_z}{\partial y} + b_z\frac{\partial b_z}{\partial z}\right), \tag{3}$$

$$\frac{\partial b_x}{\partial t} + v_x\frac{\partial b_x}{\partial x} + v_y\frac{\partial b_x}{\partial y} + v_z\frac{\partial b_x}{\partial z} - b_x\frac{\partial v_x}{\partial x} - b_y\frac{\partial v_x}{\partial y} - b_z\frac{\partial v_x}{\partial z} = \lambda\Delta b_x, \tag{4}$$

$$\frac{\partial b_y}{\partial t} + v_x\frac{\partial b_y}{\partial x} + v_y\frac{\partial b_y}{\partial y} + v_z\frac{\partial b_y}{\partial z} - b_x\frac{\partial v_y}{\partial x} - b_y\frac{\partial v_y}{\partial y} - b_z\frac{\partial v_y}{\partial z} = \lambda\Delta b_y, \tag{5}$$

$$\frac{\partial b_z}{\partial t} + v_x\frac{\partial b_z}{\partial x} + v_y\frac{\partial b_z}{\partial y} + v_z\frac{\partial b_z}{\partial z} - b_x\frac{\partial v_z}{\partial x} - b_y\frac{\partial v_z}{\partial y} - b_z\frac{\partial v_z}{\partial z} = \lambda\Delta b_z, \tag{6}$$

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0, \qquad \frac{\partial b_x}{\partial x} + \frac{\partial b_y}{\partial y} + \frac{\partial b_z}{\partial z} = 0, \tag{7}$$

here v_x, v_y, v_z are the fluid velocity components, p is the pressure, $\mathbf{g}$ is the acceleration of gravity, ρ is the density, b_x, b_y, b_z are the field magnetic

intensity components, $\lambda = \dfrac{1}{\sigma\mu}$, μ is the magnetic permittivity, σ is the electric conductivity of the medium, and $\boldsymbol{\omega}$ is the angular velocity of rotation of the layer.

Consider the characteristic scales of variation of variables in equations $(1)-$ (7): D for the vertical motion (it is assumed that D is equal to the average depth of the fluid layer $h_B(x, y, t) - Z(x, y)$), L for the horizontal motion, U for the horizontal velocity component, W for the vertical velocity component, B for the horizontal field components, H for the vertical field components, T for the time component, and P for the pressure field.

For the problem under consideration it is natural to assume that

$$\delta = \frac{D}{L} \ll 1.$$

Besides, in the continuity equation (7) the first and the second terms are of the order $O\left(\dfrac{U}{L}\right)$, and hence the order of the third term $O\left(\dfrac{W}{D}\right)$ is not greater than $O\left(\dfrac{U}{L}\right)$. Therefore,

$$W \leq O(\delta U).$$

In a similar manner, estimating the orders of the terms in the solenoidality equation (7), we find that

$$H \leq O(\delta B).$$

Estimating further the terms in (1)–(6), we change in these equations to the dimensionless variables. As a result, we have

$$\frac{U}{T}\frac{\partial v_x}{\partial t} + \frac{U^2}{L}\left(v_x\frac{\partial v_x}{\partial x} + v_y\frac{\partial v_x}{\partial y} + v_z\frac{\partial v_x}{\partial z}\right) = -\frac{1}{\rho L}\left(P + \frac{(1+\delta^2)B^2}{2\mu}\right)\cdot$$
$$\cdot\frac{\partial}{\partial x}\left(p + \frac{b^2}{2\mu}\right) + 2\omega U v_y + \frac{B^2}{L\mu\rho}\left(b_x\frac{\partial b_x}{\partial x} + b_y\frac{\partial b_x}{\partial y} + b_z\frac{\partial b_x}{\partial z}\right), \qquad (8)$$

$$\frac{U}{T}\frac{\partial v_y}{\partial t} + \frac{U^2}{L}\left(v_x\frac{\partial v_y}{\partial x} + v_y\frac{\partial v_y}{\partial y} + v_z\frac{\partial v_y}{\partial z}\right) = -\frac{1}{\rho L}\left(P + \frac{(1+\delta^2)B^2}{2\mu}\right)\cdot$$
$$\cdot\frac{\partial}{\partial y}\left(p + \frac{b^2}{2\mu}\right) - 2\omega U v_x + \frac{B^2}{L\mu\rho}\left(b_x\frac{\partial b_y}{\partial x} + b_y\frac{\partial b_y}{\partial y} + b_z\frac{\partial b_y}{\partial z}\right), \qquad (9)$$

$$\frac{\delta U}{T}\frac{\partial v_z}{\partial t} + \frac{\delta U^2}{L}\left(v_x\frac{\partial v_z}{\partial x} + v_y\frac{\partial v_z}{\partial y} + v_z\frac{\partial v_z}{\partial z}\right) = -\frac{1}{\rho D}\left(P + \frac{(1+\delta^2)B^2}{2\mu}\right)\cdot$$
$$\cdot\frac{\partial}{\partial z}\left(p + \frac{b^2}{2\mu}\right) - g + \frac{\delta B^2}{L\mu\rho}\left(b_x\frac{\partial b_z}{\partial x} + b_y\frac{\partial b_z}{\partial y} + b_z\frac{\partial b_z}{\partial z}\right), \qquad (10)$$

$$\frac{B}{T}\frac{\partial b_x}{\partial t} + \frac{UB}{L}\left(v_x\frac{\partial b_x}{\partial x} + v_y\frac{\partial b_x}{\partial y} + v_z\frac{\partial b_x}{\partial z} - b_x\frac{\partial v_x}{\partial x} - b_y\frac{\partial v_x}{\partial y} - b_z\frac{\partial v_x}{\partial z}\right) =$$
$$= \frac{\lambda B}{L^2}\Delta b_x, \qquad (11)$$

$$\frac{B}{T}\frac{\partial b_y}{\partial t} + \frac{UB}{L}\left(v_x\frac{\partial b_y}{\partial x} + v_y\frac{\partial b_y}{\partial y} + v_z\frac{\partial b_y}{\partial z} - b_x\frac{\partial v_y}{\partial x} - b_y\frac{\partial v_y}{\partial y} - b_z\frac{\partial v_y}{\partial z}\right) =$$
$$= \frac{\lambda B}{L^2}\Delta b_y, \qquad (12)$$

$$\frac{\delta B}{T}\frac{\partial b_z}{\partial t} + \frac{\delta UB}{L}\left(v_x\frac{\partial b_z}{\partial x} + v_y\frac{\partial b_z}{\partial y} + v_z\frac{\partial b_z}{\partial z} - b_x\frac{\partial v_z}{\partial x} - b_y\frac{\partial v_z}{\partial y} - b_z\frac{\partial v_z}{\partial z}\right) =$$
$$= \frac{\delta\lambda B}{L^2}\Delta b_z. \qquad (13)$$

Here and below, we retain the notation for the dimensionless variables.

In equations (8) and (9), all the terms are of the same order, provided that the dynamic, magnetic and kinetic pressures are of the same order: $P \backsim \dfrac{B^2}{\mu} \backsim$ $\rho U^2 \backsim \dfrac{\rho U}{T}$. In this case, equations (8) and (9) will be used without change for further investigation.

The ratio of the convective term in the induction equations (11)–(13) to the diffusive term, as expressed in terms of the characteristic velocity of liquid U and the characteristic length L, is the dimensionless parameter $\dfrac{LU}{\lambda}$, which is known as the magnetic Reynolds number that characterizes the relation between the plasma flow and the magnetic field. In laboratory experiments one usually has $R_m \ll 1$, and so this coupling is weak, whereas in astrophysics it is typical that $R_m \gg 1$ and this coupling is strong [1]. The induction equation controls the behavior of the magnetic field if one knows the velocity; this behavior depends

substantially on the values of the magnetic Reynolds number R_m. In general case, magnetic field line are partially transferred by plasma flow and partially diffuse through it.

In what follows we shall be considered with this general case. So, we write $R_m = 1$ and assume that the diffusion terms have the same order as the convection terms.

The consideration of the diffusion terms is necessary in the study of more local dynamics of waves; i.e., when L is much smaller than the layer radius, and also for very large time scales T. It would be interesting to observe the effect of the magnetic field diffusion on its generation. Another question is whether such a field can exist arbitrarily long and wether it would exist if the seed field is switched off.

Retaining in (10) only the principal terms, we get

$$\frac{\partial}{\partial z}\left(p + \frac{b^2}{2\mu}\right) = -\rho g.$$

Hence, integrating with respect to z,

$$p + \frac{b^2}{2\mu} = -\rho g z + C(x, y, t),$$

and further, using the boundary conditions $p(x, y, -h_B) = p_0(x, y, t)$ and $b(x, y, -h_B) = b_0(x, y, t)$, we get

$$p + \frac{b^2}{2\mu} = p_0 + \frac{b_0^2}{2\mu} - \rho g(h_B + z). \tag{14}$$

The consequence of the linear dependence on z of the right-hand side of (14) is that the horizontal gradient of hydromagnetic pressure is independent of z. As a result, the horizontal components of the velocity and of the magnetic field are also independent of z, provided that they are independent of z at the initial time.

Integrating with respect to z the solenoidality equation and the equation of continuity for the fluid, using the boundary impermeability condition and the condition specifying the magnetic field normal component on the boundary surfaces, we get a system which has fewer (when compared with the original system) number of dynamic equations, unknown functions (due to exclusion of v_z and b_z from the equations of the original system), and independent variables (because z does now not enter explicitly into the dynamic equations). The sought-for variables v_x, v_y, b_x, b_y, h_B depend only on x, y, t, while the functions v_z and b_z depend linearly on z.

Let us introduce the full depth function $H = h_B - Z = H_0(x, y) + \eta(x, y, t)$, where $\eta(x, y, t)$ is a small perturbation characterized by the inequality $\eta \ll H_0$. To describe the propagation of small perturbations, we shall employ the method of linearization of the system of differential equations that describe the behavior of the medium. This approach is pretty standard in continuum mechanics. The

solution of the corresponding system will be sought in the form

$$\mathbf{v} = \mathbf{v}_0 + \mathbf{v}'(x, y, t), \qquad \mathbf{b} = \mathbf{b}_0 + \mathbf{b}'(x, y, t), \tag{15}$$

assuming that the small perturbations of the horizontal velocity $\mathbf{v}'$ and of the horizontal magnetic field $\mathbf{b}'$ propagate over some stationary uniform background described by the constant terms $\mathbf{v}_0$, $\mathbf{b}_0$, where $\mathbf{v}_0 = 0$.

We now consider the problem with due account of perturbations of first order of smallness. Let us introduce into consideration the modified perturbation functions of the boundary surface and of the horizontal components of the magnetic field, which are defined as

$$\eta(x, y, t) = \frac{1}{g} \mathcal{D}_t \left(\mathcal{D}_t^2 + \alpha^2 \right) \widetilde{\eta}(x, y, t), \tag{16}$$

$$b_x(x, y, t) = \mu\rho \mathcal{D}_t \left(\mathcal{D}_t^2 + \alpha^2 \right) \widetilde{b}_x(x, y, t), \tag{17}$$

$$b_y(x, y, t) = \mu\rho \mathcal{D}_t \left(\mathcal{D}_t^2 + \alpha^2 \right) \widetilde{b}_y(x, y, t), \tag{18}$$

$$P(x, y, t) = \mathcal{D}_t \left(\mathcal{D}_t^2 + \alpha^2 \right) \widetilde{P}_y(x, y, t), \tag{19}$$

where $\mathcal{D}_t = \dfrac{\partial}{\partial t}$. For $H_0 = \text{const}$ and $\nabla Z = \text{const}$, at distances of the order of the wavelength, the original boundary-value problem reduces to the equation

$$\mathcal{D} \left(\mathcal{D}_t^2 + \alpha^2 \right)^2 \left[\left(\mathcal{D}_t - \frac{\Delta}{R_m} \right) \mathcal{D}_t - \frac{\mathcal{D}^2}{\mu\rho} \right] \Delta_2 \xi + \frac{1}{g(\mu\rho)^2 H_0} \left(\mathcal{D}_t^2 + \alpha^2 \right) \times$$

$$\times \left[F^2 + \left(\alpha \mathcal{D}^2 \right)^2 \right] \mathcal{D}\xi = -\frac{b_{n0}^{(e)}}{(\mu\rho)^2 H_0} - \mathcal{D} \left(\mathcal{D}_t^2 + \alpha^2 \right)^2 \left[\left(\mathcal{D}_t - \frac{\Delta}{R_m} \right) \mathcal{D}_t - \frac{\mathcal{D}^2}{\mu\rho} \right] \Delta_2 P,$$

which has a solution as a harmonic wave with two distinctly separate branches for the frequency. The first type of oscillations is given by inertial waves, in which the inertia and the Coriolis force play an important role. The frequency of inertial waves is real; these waves are stable. The second type of oscillations comes from the waves due to the combined action of magnetic forces, the gravitational force, the Coriolis force, and boundary effects. In general, their frequency can be complex, and so these waves can be unstable.

In particular, in the case of a frozen magnetic field, the boundary effects considered above may promote an instability, which in turn is responsible for the growth of the magnetic field. At the same time, by controlling the values of the seed magnetic field, one can observe a time-steady process—this means that the induced magnetic field can exist arbitrarily long. Moreover, in the absence of the external magnetic field and under certain relation between the influencing parameters several cases are possible: a steady-state wave regime may preserve, an unstable wave regime may appear, or the wave process may decay.

References

1. Alfvén, H., Fälthammar, C.-G.: Cosmic Electrodynamics. Oxford, Clarendon (1963).

Development and research of computer vision algorithms for visual control of geometric parameters of objects (defining the boundaries of the contour of the part)

E. Ryzhkova, O. Baklanova, A. Baklanov, and M. Pronina

D.Serikbayev East Kazakhstan State Technical University, School of Information
Technology and Energy,
Serikbayev street 19, 070010 Ust-Kamenogorsk, Kazakhstan
`alena.tarasova.2014@mail.ru`

Abstract The article deals with image processing techniques using machine vision, the ability to replace visual quality control of technological products in industries requiring high-precision measurements. The article represents the determination of the boundaries of a titanium implant pin using the Sobel's method of the MATLAB program. This method is based on the search for maxima, allocates boundaries using the "edge strength"calculation, usually the expression of the first derivative, such as the magnitude of the gradient, and then the search for local maxima of the edge strength, using the assumed direction of the boundary, usually perpendicular to the gradient vector.

Keywords: digitalization, machine vision, image recognition, image processing methods, quality control.

Introduction

The Government program «Digital Kazakhstan» for 2017-2020 [1] is aimed at the development and digital transformation in the economic sectors - development of the digital industry through the automation of the transport and logical system of the country, the introduction of digital technologies in the field of industry; ensuring the availability of digital information; the implementation of technologies for the creation of smart cities. The importance of digitization is attached by the President of the Republic of Kazakhstan N.Nazarbayev: "I announced in my Message to the people of Kazakhstan about the Third Modernization, the core of which is digitalization."[2]

In connection with the development of industry, in our time the issue of high-precision testing of product quality with the help of new technologies has become urgent. Machine vision allows you to save, record images, shape, size, location position, as well as the texture of the product. Visual inspection systems with machine vision have a high speed of operation, the possibility of 24-hour

operation and the accuracy and reliability of measurements. The advantage of machines over a person is the absence of fatigue, illness or inattention.

Use of machine vision provides the following options:

1. The machine vision monitoring system will allow to exclude the human factor in the visual inspection of the quality of products, which is of no small importance in the manufacture of precision products, namely the production of medical materials and products;

2. The machine vision monitoring system can be used as a technology to assess the geometric parameters of an object, the presence of defects on the contours of the part, on the thread, which is important for carrying out work in applied materials research, metallurgy and obtaining medical materials and products.

Machine vision is directly engineering direction, it includes digital input-output devices and computer networks designed to control production equipment, robotic manipulators or devices for extracting defective products. Machine vision is a direction in engineering related to computer technology, optics, mechanical engineering and industrial automation. One of the most common applications of machine vision is the inspection of goods, such as semiconductor chips [3], cars [4], food [5] and medicines [6]. Machine vision control systems for product testing use digital and intelligent cameras, software [7] for image processing proposed in and performing similar checks.

The article considers the use of computer vision techniques to control the quality of dental implants, in this case the main part for control is a pin or a screw, which is the root part inserted into the jaw tissue of the patient, its integrity depends on the installation, implantation and further operation of the implant.

Methods of machine vision for visual control of the geometric parameters of objects (defining the boundaries of the contour of the part)

The usage of computer vision techniques is promising for the quality control of implants in medicine precisely in traumatology and dentistry. Since endoprosthetics is an effective method of treating and restoring the integrity of bone tissue. Implants can reduce the duration of treatment for severe diseases and exclude rejection of the latter. The quality of the initial material and the quality of the deposition during endoprosthetics are important in the manufacture of the implant. The use of machine vision will replace the visual quality control of medical implants on the machine. This will eliminate the human factor in identifying the marriage, speed up the production process, reduce labor costs, therefore reduce the cost of the finished product and, most importantly, improve the quality of the products.

We will consider a computer vision quality monitoring system that detects defects associated with the geometric characteristics of the object. In particular,

cylindrical parts and details of complex shape (in our case with a thread). Consider some types of defects:

Figure 1 shows the defect of the implant carving in the form of a crack. Cracks are fractures that occur at boundaries or inside crystals, as well as at the location of nonmetallic inclusions as a result of metal over-stressing during processing. Cracks in turn are divided into: shear cracks, stamping, cracks from scratches, from hot bubbles.

Figure 1. Defect of the implant carving in the form of a fracture.

Figure 2 depicts a rags defect, flaws - are open gaps in the metal. Occur in places that are most deformed: the surface and the edge.

Figure 2. Defect is a rags.

Figure 3 shows the defect - folds. Folds - crushed metal protrusions, formed during the process of stamping the product or rolling the thread. Are located in places of change of a cross-section of an article or on a thread profile.

Figure 4 depicts the defect obtained as a result of inadequate use of tools. Tool marks: longitudinal and annular risks arising from the movement of the tool over the surface of the product.

Figure 3. Defect - folds.
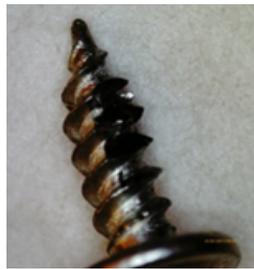
Figure 4. Defect - marks from tools.

Figure 5 shows a defect in the form of flash. Flash - this is a defect in the surface of the thread, which is a sharp, in the form of a ridge, protrusion, formed when cutting.

Figure 5. Defect is a flash.

The article presents the possibilities of machine vision:

1. Using the Sobel method to recognize contours, and using filters to get a sharper image. This allow us to apply machine vision to detect defects in the boundaries of the pin products, i.e. directly its threads;
2. Using the MATLAB program to define geometric parameters and contours and output data.

Theoretical Part

There are many approaches to the allocation of boundaries. They can be divided into two categories: methods based on the search for maxima, and methods based on the search for zeros. Methods based on the search for maxima distinguish boundaries by calculating the "edge strength". These are expressions of the first derivative of the magnitude of the gradient and then the search for local maxima of the edge strength. Using the assumed direction of the boundary, it is usually perpendicular to the gradient vector. Methods based on the search for zeros: they search for the intersection of the abscissa axis with the second derivative, usually the zeros of the Laplacian [8] or the zeros of the nonlinear differential expression, as will be described later. As a preprocessing step, borders are almost always applied to image smoothing, usually by a Gaussian filter [9]. The published methods for delineating boundaries are distinguished by the applied anti-aliasing filters and methods, as is the edge strength. Although many border allocation methods are based on the calculation of an image gradient, they differ in the types of filters used to calculate the gradients in the x- and y-directions. From the point of view of image recognition, the most informative are the outlines of objects, that is, their boundaries. The carrier of information is not the brightness, but the boundary of the objects in the image.

The Sobel operator [10] is the discrete differential operator calculating approximate value of a gradient of brightness of the image. Application of the operator Sobel in each point of the image is or a vector of a gradient of brightness in this point, or his norm. Used in the field of image processing is often used in algorithms for delimiting boundaries. The Sobel operator is based on the convolution of images by small separable integer filters in vertical and horizontal directions, so it is relatively easy to compute. On the other hand, the gradient approximation is rather rough - this affects the high-frequency oscillations of the image. The operator calculates the gradient of the brightness of the image at each point. This is the direction of the greatest increase in brightness and its magnitude in this direction. The result is how sharply or "smoothly"the brightness of the image changes at each point, and hence the probability of finding a point on the face, and also the orientation of the boundary. In practice, the calculation of the magnitude of the change in brightness (the probability of belonging to the face) is more reliable and easier to interpret than the calculation of the direction. Mathematically, the gradient of a function of two variables for each image point (which is the brightness function) is a two-dimensional vector, the components of which are the derivatives of the image brightness along the horizontal and vertical lines. At each point of the image, the gradient vector is oriented in the direction of the greatest increase in brightness, and its length corresponds to the magnitude of the brightness change. This means that the result of the Sobel operator at the point lying in the region of constant brightness is the zero vector, and at the point of the region of different brightness lying on the boundary, a vector crossing the boundary in the direction of increasing brightness. The operator uses 3x3 kernels with which the original image is folded to calculate the approximate values of the derivatives horizontally and vertically.

Let A be the original image, and Gx and Gy be two images on which each point contains approximate derivatives with respect to x and y. They are calculated as follows:

$$\mathbf{Gx} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \tag{1}$$

and

$$\mathbf{Gy} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \tag{2}$$

Where (*) denotes a two-dimensional convolution operation.

The coordinate x here increases "to the right and y - "down". At each point of the image, the approximate value of the gradient value can be calculated by using the approximate values of the derivatives obtained:

$$G = \sqrt{G_x^2 + G_y^2} \tag{3}$$

Using this information, we can also calculate the direction of the gradient:

$$\theta = arctan \frac{G_y}{G_x} \tag{4}$$

where, for example, the angle θ is equal to zero for the vertical boundary, which has the dark side to the left.

Since the brightness function is known only at discrete points, we can not determine the derivatives until we set the brightness of the differentiable function that passes through these points. With this additional premise, the derivative of a differentiable luminance function can be calculated both from the function with which measurements are taken-the points of the image. It turns out that the derivatives at any single point are brightness functions from all points of the image. However, the approximations of their derivatives can be determined with a greater or lesser degree of accuracy. The Sobel operator represents a more inaccurate approximation of the gradient of the image, but it is sufficiently qualitative for practical application in many problems. More precisely, the operator uses intensity values only in the neighborhood of 3 × 3 of each pixel to obtain an approximation of the corresponding image gradient, and uses only integer values of the luminance weight coefficients to estimate the gradient.

The software implementation of the Sobel operator can effectively use the SIMD-extensions of the command system of modern processors (code vectorization), while the gain in the calculation speed of the operator can be up to five times compared with the high-level implementation. Hand coding in assembly language allows you to overtake such compilers as Microsoft Visual C++ and Intel C++ Compiler. [11]

The calculation of the Sobel operator is elementary parallel to an arbitrary number of flows (In the limit, each point of the resulting image can be calculated independently of neighboring). For example, in the presence of two processors (cores), the upper half-frame of the image can be processed by one of them, and the bottom one by another.

For example, in the presence of two processors (cores), the upper half-frame of the image can be processed by one of them, and the bottom one by another. As an object, an implant was chosen which was a screw made of titanium, the image of the implant was loaded into the MATLAB program. After this, the image was transformed into a halftone image, which is shown in Fig1. Using the ability of the MATLAB library, the code for using the Sobel transform was written. As a result of the work of the program, an implant contour was identified according to which the defects can be identified in the future as a result of its manufacture. The resulting image of the implant contour is shown in Fig 2.

Practical Part

Using MATLAB R2014b, Sobel's masks [12] were applied to determine geometric parameters and output data loops. As an object, an implant of the screw, was made into the MATLAB R2014b program. Next, with the Image Processing Toolbox, the execution of long-term encoding and debugging algorithms, allows you to focus on solving the main scientific or practical problem. The main structural unit of data in MATLAB is an array. To represent images, the data in the arrays must carry information about the corresponding values of the color intensities. MATLAB remembers most images as a two-dimensional array (matrixes), in which each element corresponds to one pixel of the displayed image. (The name of the pixel "pixel"is short for "picture element".) For example, if the image consists of 200 rows and 300 columns, it is stored in the MATLAB system as a matrix with a dimension of 200×300. This representation of images simplifies their processing and allows the full use of possibly MATLAB systems, in particular, in the field of image processing. [13]

Halftone image is often used in practice in those cases when the conditional difference between the brightness of one part of the image from another is not enough. Therefore, after loading the image of the medical pin or screw into the program MATLAB R2014b, the image was transformed into a halftone image in the format double which is shown in Fig 1. For a set of possible halftones are called gray levels (Eng. gray scale), regardless of whether a semitone of what color or its hue is transmitted. Just as a binary image, often called "black and white,"may look "black and green"when rendered. Thus, the gray levels do not differ in spectral composition (shade of color), but differ in brightness. The number of possible halftones in this case is the color depth, which is often transmitted not in the number of halftones themselves, but in the number of bits per pixel (Eng. bit per pixel, bpp). Which of the values in the acceptable range will be considered the brightest, and which is the darkest does not matter. since

the number being the value of each pixel is just a conditional brightness code. It is enough to specify the direction of counting.

For example, there may exist halftone rasters, where 8 bits are allocated for each pixel, the image has 256 semitones, and pixels with a value of 0 or 255 are black, and vice versa, pixels with a value of 255 or 0 are white, the remaining gray halftones will be evenly distributed between these values color index. [15]
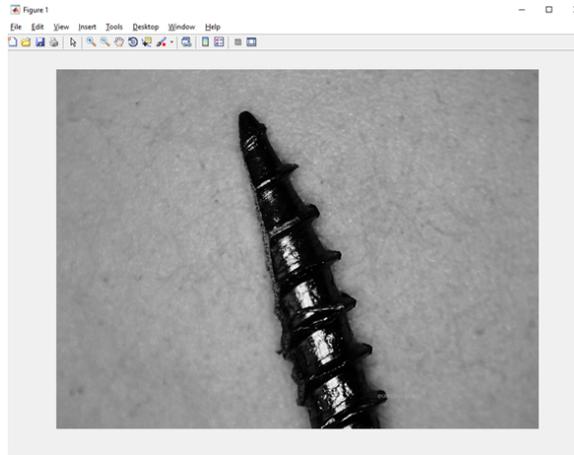
Figure 6. Halftone image in double format in the program MATLAB R2014b.

The calculation of the Sobel operator is elementary parallelized to an arbitrary number of streams (in the limit, each point of the resulting image can be calculated independently of neighboring ones).

The Sobel operator consists of two separate operations [14]:

1. Smoothing with a triangular filter in a direction perpendicular to the derivative;
2. Finding a simple central change in the direction of the derivative.

The third stage of image processing is smoothing by a triangular filter in a direction perpendicular to the derivative and finding a simple central change in the direction of the derivative, which gives the normalized image shown in Figure 7.

Consider the selection of boundaries by the Sobel method. The original image is uploaded to MATLAB R2014b, then converted to grayscale in order to increase the difference in brightness of one section of the image from the other and is shown in Figure 6. The result is a normalized image, on which borders are marked, is shown in Figure 7.

In MATLAB R2014b for image processing, the function hy=fspecial('sobel'); labergb12 - this is a data transformation. The selection of borders by the Sobel method and the output of the result on the screen.
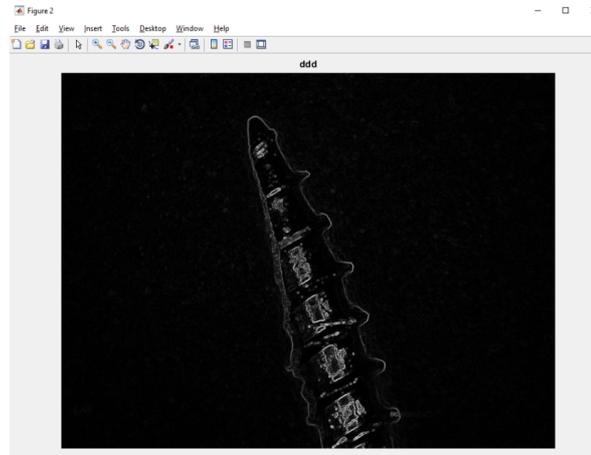
Figure 7. The normalized image of the screw in the MATLAB R2014b program.

Then we form a two-dimensional array of mismatches. The points of difference of a pair of images are determined, a two-dimensional mask of discrepancies is formed and primary filtration of small spots is carried out. Forming and filtering the list of mismatch areas. A transition is made from the original representation of the data, as a two-dimensional array of pixels, to the list of descriptions of each individual area. The generated list of areas is subjected to successive stages of analysis, filtration, and simplification. As a result of the above actions, the obtained image is compared with the reference image (comparison operation K = imlincomb (.1,2)); then a discrepancy (defect) or conformity.

Conclusion

In this article we consider image processing based on the MATLAB program that allows you to create a data library with the content of the product standard in this pin case and automatically the program will sort out the products. This will automate product quality control with the help of computer vision using (computer, MATLAB program, a system of selected image transformation methods). All this will make it possible to automate the process of controlling the quality of products at the production stages.

Acknowledgment

industrial enterprises of East Kazakhstan region"for 2017-2019, under the subprogramme 0006/PCF "Production of titanium products for further use in medicine".

References

1. State program "Digital Kazakhstan"for the years 2017-2020 `https://zerde.gov.kz/images`
2. Speech by the Head of State N.A. Nazarbayev `https://ru.sputniknews.kz/`
3. Machine vision `http://neuronus.com/`
4. «Machine vision»: what and how to see cars `https://geektimes.ru/`
5. Computer vision in agriculture `http://cosmoport.club/post/`
6. Overview of computer vision in medicine `https://habrahabr.ru/`
7. Grigoryev, Y.A., Sayun, V.M., Grigoryeva, S.V., Titov, D.N. Study of illumination properties of high-power LEDs in various temperature conditions, International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices, EDM, pp. 309-313 (2013).
8. Duda R., Hart P. Pattern Classification and Scene Analysis. — John Wiley and Sons — P. 271—272. (1973)
9. Vatutin EI, Miroshnichenko S.Yu., Titov V.S. Software optimization of the operator Sobela using SIMD extensions of the x86 family. Telecommunications. 2006. № 6. P. 12-16. (2006).
10. Mark S. Nixon and Alberto S. Aguado. Feature Extraction and Image Processing. — Academic Press pp. 88. (2008)
11. Parker, James R. Algorithms for Image Processing and Computer Vision. New York: John Wiley & Sons, Inc. (1997)
12. The competence center of the MATLAB program `http://matlab.exponenta.ru/imageprocess/book5/6_1.php`
13. Conversion of a Color Image to a Binary Image, CoderSource.net (2008)
14. `https://lektsii.org/9-75432.html`

Mathematical modeling of conveyor data processing technology in heat-networks

I. Sagynganova, O. Baklanova, and A. Baklanov

D. Serikbayev East Kazakhstan State Technical University
Department of Instrument Engineering and Technology Process Automation
Protozanov 69, 070004 Ust-Kamenogorsk, Kazakhstan
ABaklanov_62@mail.ru

One of the up-to-date areas of data processing is the development of algorithms and technologies that allow you to monitor and to manage data coming from a large number of sources. One of the possible options for data processing is conveyor parallel data processing which is implemented as standard functions in modern programming languages and allows you to work simultaneously with data stored in the computer and to process the data of many computers that are integrated into a single network. However, in real conditions we have a situation where the data server is sufficiently remote from the peripheral equipment and the users' computers. This situation results in the increase of data processing time as the amount of incoming data increases too. In such way the rapid data processing which influences on the safety and operating mode of peripheral devices can be delayed. To improve the efficiency of work with a large data thread and to organize the priorities for work with data it is proposed to use the technology of the work of the central processing unit (CPU) with the threads for the work of the computer with data coming from the network. The organization of work of the central processor with threads is considered to be successful for the personal computer. This type of organization is an advantage for work with real data.

To understand the interaction mechanism of the processor with threads it is necessary to give definitions and information related to threads and processes [1,2].

A thread in Windows is an object of the core. The operating system gives it CPU time for executing applications. Each thread includes the following resources:
- a code of the executable function;
- a set of processor registers;
- a stack for the application;
- a stack for operating system operation;
- an access token that includes information for the security system.

All these resources form the context of the thread in Windows. In addition to the descriptor each thread in Windows also has its own identifier which is unique for threads implemented in the system. The identifiers of the threads are used by service programs that allow users of the system to monitor the operation of threads.

In operating systems of Windows two types of threads can be determined:

- system threads;
- users' threads.

System threads execute various operating system services, and they are started by the operating system core. Users' threads are used to solve users' tasks, and they are started by the application. In our case we will consider users' threads. "Simultaneous"or "conveyor-parallel"operation of threads implies that if there is only one processor with one core, only one thread can be executed every moment of time. However, the operating system can quickly switch the processor from one thread to another and the user has the illusion of the simultaneous operation of several programs due to the high frequency of the processors. The operating system usually requires 0.6 ns for the thread processing. This situation is called a "pseudo-parallel" or "conveyor-parallel"operation of threads. You can see the conveyor-parallel operation of threads in Figure 1. In the operation of the processor with threads (processes) the ability to define the priorities of threads processed by the processor is important. The higher the priority the more time is required for processing this thread.
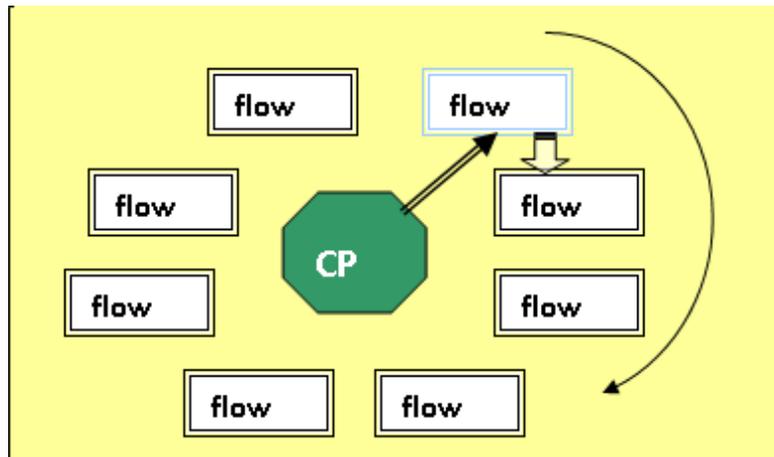
Figure 1. The time interval required by operating system

We will consider the possibility of using this technology for the data processing of a heat point. The main tasks of heat points are [3]:
- the transformation of the heat carrier;
- the control and regulation of the heat carrier parameters;
- the distribution of the heat carriers by heat consumption systems;
- the disconnection of heat consumption systems;
- the protection of heat consumption systems from emergency increase of the heat carrier parameters;
- the account of heat carriers and heat consumption.

Obviously, when carrying out these tasks, it is necessary to take into account the priority of the task and to work not with one heat point, but with a number of heat points that are serviced by the central heat point. It's easy to implement the standard functions used with the CPU to work with processes:
An example of a computer program

```
BOOL CreateProcess
LPCTSTR lpApplicationName,     // name of executable module
LPTSTR lpCommandLine,          // command line
LPSECURITY_ATTRIBUTES lpProcessAttributes, // SD (security
                                           // descriptor)
LPSECURITY_ATTRIBUTES lpThreadAttributes,  // SD
BOOL bInheritHandles,          // descriptor of inheritance
                               // parameter
DWORD dwCreationFlags,         // flags of creation
LPVOID lpEnvironment,          // new block of configuration
LPCTSTR lpCurrentDirectory,    // name of current directory
LPSTARTUPINFO lpStartupInfo,   // preset information
LPPROCESS_INFORMATION lpProcessInformation // process
                                           // information
```

In our case the LPCTSTR executable lpApplicationName module performs the maintenance of the heat point, i.e. you can start the required number of processes that will monitor the operation of all the heat points involved in the service of the central heat point. It is very important that there is a possibility to ensure the safety of the operation of the heat point by installing the required safety descriptor both for the operation of the definite heat point and for the thread which is responsible for a particular task.

Let's consider in more details the work of the proposed methodology for data processing of heat points. The scheme of operation of the system is shown in Figure 2.

The main idea is that a process is created for the maintenance of each heat point. It consists of 6 threads. Each thread corresponds to the task performed by the heat station. So the thread No.1 corresponds to the transformation of the heat carrier type, the thread №2 corresponds to the control and regulation of the heat carrier parameters, the thread №3 - the distribution of the heat carriers by the heat consumption systems, the thread №4 - the disconnection of the heat consumption systems, the thread №5 - the protection of the heat consumption systems from the emergency increase of the heat carrier parameters, the thread number № 6 – the account of heat carriers and heat consumption.

To understand the operation of a personal computer with heat points we need to consider the function of creating a thread in the Windows operating system.

When you start a new process using the CreateProcess function, the thread identifier is stored in the dwThreadld block of the PROCESS_INFORMATION structure which is activated after the return of the function call. The thread
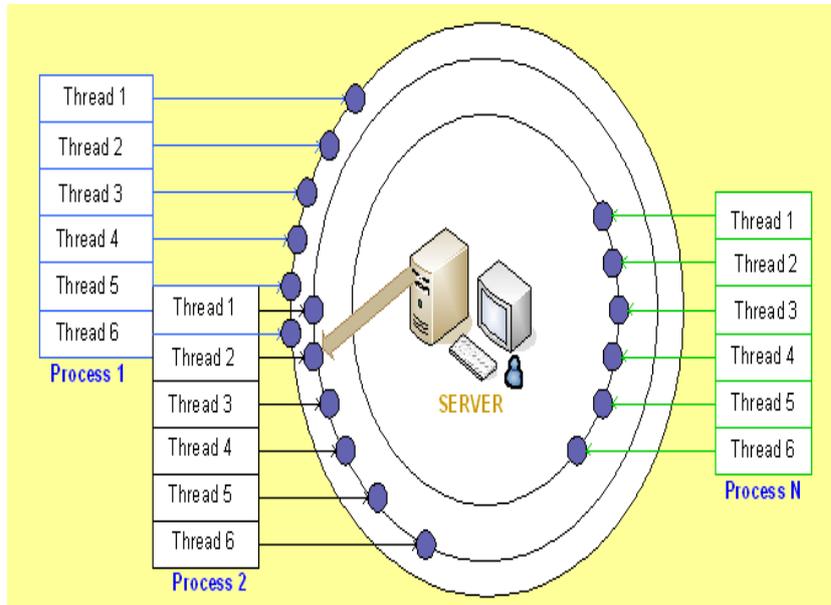
Figure 2. The scheme of data processing by a personal computer

reference is in the hThread block of the same structure. The thread is created as a result of the function called CreateThread:

```
Create Thread
        NULL,          // default security attributes
        0,             // use default stack size
        ThreadFunc,    // thread function
        &dwThrdParam,  // argument to thread function
        0,             // use default creation flags
        &dwThreadId);  // returns the thread identifier
```

In the proposed technology of using processes and threads for the data processing of heat points the ThreadFunc parameter reflects a certain task performed by the heat point.

It should be noted the possibility of introducing priorities in the data processing coming from the heat points. This information refers to the creation of the priority for a certain heat point and the priority of a certain task when processing data coming from the heat point. It can be performed by analogy with the priorities organization when operating with processes and threads. So the priority class of the process is changed by the SetPriorityClass function call:

BOOL SetPriorityClass (HANDLE hProcess, DWORD fdwPriority);

This function changes the priority class of the process defined by the hProcess handle according to the value of the fdwPriority parameter. The latter must contain one of the values according to the priority. So far as SetPriorityClass

accepts a process handle you can change the priority of any process implemented in the system if its descriptor is known and has the appropriate access rights.

To change the priority of the primary thread we use the function:

BOOL SetThreadPriority(HANDLE hThread, int nPriority);

Obviously, the hThread parameter points to a thread which is changed. The proposed methodology will allow to ensure high speed of data processing of heat points as well as to increase the reliability and safety due to the introduction of the priority of threads execution.

References

1. Pobegailo, A.: System programming in Windows, St. Petersburg: (2006) p 1056
2. Tapenambum, E.: Modern operating systems, 2nd ed. St. Petersburg/ Peter (2007) p1038
3. Pyrkov, V.: Modern thermal points. Automation and regulation. II DP "Taki spravi"(2007) p252

Содержание

206