

Computational Mathematics and Information Technologies

Computational
Mathematics

Mathematical
Modelling

Information
Technologies





Computational Mathematics and Information Technologies

Peer-reviewed scientific and theoretical journal (published since 2017)

eISSN 2587-8999

DOI: 10.23947/2587-8999

Vol. 8, no. 2, 2024

The scope of “Computational Mathematics and Information Technologies” is focused on fundamental and applied research according to the following scientific sections:

1. Computational Mathematics
2. Mathematical Modelling
3. Information Technologies

Indexing

RSCI, Crossref, Cyberleninka

Name of the body that registered the publication

Mass media registration certificate ЭЛ № ФС 77-66529 dated July 21, 2016
issued by the Federal Service for Supervision of Communications, Information
Technology and Mass Media

Founder and publisher

Federal State Budgetary Educational Institution of Higher Education Don State
Technical University (DSTU)

Periodicity

4 issues per year

Address of the founder and publisher

1, Gagarin sq., Rostov-on-Don, 344003, Russian Federation

E-mail

CMIT-EJ@yandex.ru

Telephone

+7(863) 273–85–14

Website

<https://cmit-journal.ru>

Date of publication

28.06.2024



Computational Mathematics and Information Technologies

Рецензируемый научно-теоретический журнал (издаётся с 2017 года)

eISSN 2587-8999

DOI: 10.23947/2587-8999

Том 8, № 2, 2024

Журнал «Computational Mathematics and Information Technologies» ориентирован на фундаментальные и прикладные исследования по следующим научным разделам:

1. Вычислительная математика
2. Математическое моделирование
3. Информационные технологии

<i>Индексация</i>	РИНЦ, CrossRef, CyberLeninka
<i>Наименование органа, зарегистрировавшего издание</i>	Свидетельство о регистрации средства массовой информации ЭЛ № ФС 77 – 66529 от 21 июля 2016 г., выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций
<i>Учредитель и издатель</i>	Федеральное государственное бюджетное образовательное учреждение высшего образования «Донской государственный технический университет» (ДГТУ)
<i>Периодичность</i>	4 выпуска в год
<i>Адрес учредителя и издателя</i>	344003, Российская Федерация, г. Ростов-на-Дону, пл. Гагарина, 1
<i>E-mail</i>	CMIT-EJ@yandex.ru
<i>Телефон</i>	+7(863) 273–85–14
<i>Сайт</i>	https://cmit-journal.ru
<i>Дата выхода в свет</i>	28.06.2024

Editorial Board

Editor-in-Chief, Alexander I. Sukhinov, Corresponding member of RAS, Dr.Sci. (Phys.-Math.), Professor, Don State Technical University (Rostov-on-Don, Russia), [MathSciNet](#), [eLibrary.ru](#), [ORCID](#), [ResearcherID](#), [Scopus](#), sukhinov@gmail.com, spu-40.4@donstu.ru

Deputy Chief Editor, Mikhail V. Yakobovskii, Corresponding Member of RAS, Dr.Sci. (Phys.-Math.), Professor, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia), [eLibrary.ru](#), [ORCID](#)

Executive Secretary, Alexander P. Petrov Dr.Sci. (Phys.-Math.), Head Scientist Researcher, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia), [eLibrary.ru](#), [Istina](#), [ORCID](#), [ResearcherID](#), [Scopus](#)

Vladimir V. Voevodin, Corresponding Member of RAS, Dr.Sci. (Phys.-Math.), Professor, Lomonosov Moscow State University (Moscow, Russia)

Vladimir A. Gasilov, Dr.Sci. (Phys.-Math.), Professor, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia)

Valentin A. Gushchin, Corresponding Member of RAS, Dr.Sci. (Phys.-Math.), Professor, Institute of Computer Aided Design, Russian Academy of Sciences (Moscow, Russia)

Galina G. Lazareva, Corresponding member of RAS, Dr. Sci. (Phys.-Math), Professor of RAS, RUDN University, (Moscow, Russia)

Vladimir I. Marchuk, Dr.Sci. (Eng.), Professor, Don State Technical University (Rostov-on-Don, Russia)

Igor B. Petrov, Corresponding Member of RAS, Dr.Sci. (Phys.-Math.), Professor, Moscow Institute of Physics and Technology (State University) (Moscow, Russia)

Sergey V. Polyakov, Dr.Sci. (Phys.-Math.), Professor, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia)

Vladimir F. Tishkin, Corresponding Member of RAS, Dr.Sci. (Phys.-Math.), Professor, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia)

Boris N. Chetverushkin, Academician of RAS, Dr.Sci. (Phys.-Math.), Professor, Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow, Russia)

Alexander E. Chistyakov, Dr.Sci. (Phys.-Math.), Professor, Don State Technical University (Rostov-on-Don, Russia)

Редакционная коллегия

Главный редактор, Сухинов Александр Иванович, член-корреспондент РАН, доктор физико-математических наук, профессор, Донской государственный технический университет (Ростов-на-Дону, Россия), [MathSciNet](#), [eLibrary.ru](#), [ORCID](#), [ResearcherID](#), [Scopus](#), sukhinov@gmail.com, spu-40.4@donstu.ru

Заместитель главного редактора, Якобовский Михаил Владимирович, член-корреспондент РАН, доктор физико-математических наук, профессор, Институт прикладной математики им. М.В. Келдыша РАН (Москва, Россия), [eLibrary.ru](#), [ORCID](#)

Ответственный секретарь, Петров Александр Пхоун Чжо, доктор физико-математических наук, ведущий научный сотрудник, Институт прикладной математики им. М. В. Келдыша РАН (Москва, Россия), [eLibrary.ru](#), [ИСТИНА](#), [ORCID](#), [ResearcherID](#), [Scopus](#)

Воеводин Владимир Валентинович, член-корреспондент РАН, доктор физико-математических наук, профессор, Московский государственный университет им. М. В. Ломоносова (Москва, Россия)

Гасилов Владимир Анатольевич, доктор физико-математических наук, профессор, Институт прикладной математики им. М. В. Келдыша РАН (Москва, Россия)

Гущин Валентин Анатольевич, член-корреспондент РАН, доктор физико-математических наук, профессор, Институт автоматизации проектирования РАН (Москва, Россия)

Лазарева Галина Геннадьевна, член-корреспондент РАН, доктор физико-математических наук, профессор РАН, Российский университет дружбы народов (Москва, Россия)

Марчук Владимир Иванович, доктор технических наук, профессор, Донской государственный технический университет (Ростов-на-Дону, Россия)

Петров Игорь Борисович, член-корреспондент РАН, доктор физико-математических наук, профессор, Московский физико-технический институт (государственный университет) (Москва, Россия)

Поляков Сергей Владимирович, доктор физико-математических наук, старший научный сотрудник, Институт прикладной математики им. М. В. Келдыша РАН (Москва, Россия)

Тишкин Владимир Федорович, член-корреспондент РАН, доктор физико-математических наук, профессор, Институт прикладной математики им. М. В. Келдыша РАН (Москва, Россия)

Четверушкин Борис Николаевич, академик РАН, доктор физико-математических наук, профессор, научный руководитель Института прикладной математики им. М. В. Келдыша РАН (Москва, Россия)

Чистяков Александр Евгеньевич, доктор физико-математических наук, профессор, Донской государственный технический университет (Ростов-на-Дону, Россия)

Contents

Congratulations to Corresponding Member of RAS Mikhail Vladimirovich Yakobovskiy	7
---	----------

COMPUTATIONAL MATHEMATICS

An Adaptive Mesh Refinement Solver for Regularized Shallow Water Equations	9
<i>I.I. But, M.A. Kiryushina, S.A. Elistratov, T.G. Elizarova, A.D. Tiniakov</i>	

MATHEMATICAL MODELLING

Probabilistic Analysis of Heat Flux Distribution in the North Atlantic for 1979–2022	24
<i>K.P. Belyaev, A.A. Kuleshov, A.V. Novikova, N.P. Tuchkova</i>	
Mathematical Modelling of Catastrophic Surge and Seiche Events in the Azov Sea Using Remote Sensing Data	33
<i>E.A. Protsenko, N.D. Panasenko, S.V. Protsenko</i>	
Modelling of Capillary Discharge in Repetition Mode for Short Capillary Systems with Various Filling Methods	45
<i>V.A. Gasilov, N.O. Savenko, Yu.S. Sharova</i>	

INFORMATION TECHNOLOGIES

Locating the Interface between Different Media Based on Matrix Ultrasonic Sensor Data Using Convolutional Neural Networks	60
<i>A.V. Vasyukov</i>	
Application of Neural Networks to Solve the Dirichlet Problem for Areas of Complex Shape	68
<i>A.V. Galaburdin</i>	

Содержание

Поздравление с юбилеем члена-корреспондента РАН М.В. Якобовского	7
--	---

ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА

Решатель с адаптивным измельчением сеток для регуляризованных уравнений мелкой воды	9
<i>И.И. Бут, М.А. Кирюшина, С.А. Елистратов, Т.Г. Елизарова, А.Д. Тиняков</i>	

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Вероятностный анализ распределения потоков тепла в Северной Атлантике за 1979–2022 годы	24
<i>К.П. Беляев, А.А. Кулешов, А.В. Новикова, Н.П. Тучкова</i>	
Математическое моделирование катастрофических сгонно-нагонных явлений Азовского моря с использованием данных дистанционного зондирования	33
<i>Е.А. Проценко, Н.Д. Панасенко, С.В. Проценко</i>	
Моделирование капиллярного разряда в режиме повторения для коротких капиллярных систем при различных способах заполнения	45
<i>В.А. Гасилов, Н.О. Савенко, Н.О. Шарова</i>	

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Определение границы раздела сред по трёхмерным данным матричного ультразвукового датчика с использованием свёрточных нейронных сетей	60
<i>А.В. Васюков</i>	
Применение нейронных сетей для решения задачи Дирихле для областей сложной формы	68
<i>А.В. Галабурдин</i>	

ANNIVERSARY OF THE SCIENTIST ЮБИЛЕЙ УЧЕНОГО



On April 8, 2024, Mikhail Vladimirovich Yakobovskiy, the Deputy Editor-in-Chief of our journal and a corresponding member of the Russian Academy of Sciences, turned 60 years old.

In Commemoration of the Anniversary of Corresponding Member of the Russian Academy of Sciences, Doctor of Physical and Mathematical Sciences, Professor Mikhail Vladimirovich Yakobovskiy

Mikhail Vladimirovich Yakobovskiy, Deputy Director for Research at the Keldysh Institute of Applied Mathematics RAS, is a leading expert in the development of parallel algorithms and simulation tools for fundamental and applied problems of continuum mechanics on high-performance computing systems. M.V. Yakobovskiy is the author of more than 90 scientific works, including 8 officially registered software programs. His scientific interests lie in the development of parallel algorithms and software for solving continuum mechanics problems on high-performance multiprocessor and hybrid computing systems, including fault-tolerant numerical simulation algorithms on exascale supercomputers.

M.V. Yakobovskiy proposed a model of a unified computing environment for solving a wide range of current fundamental and applied problems using grids containing billions or more nodes. The software suite developed based on this model integrates applications supporting the main stages of computational experiments and ensures the coordinated use of multiple distributed computing resources and clusters.

Using a unified approach to processing large volumes of grid data and the close integration of computational mathematics and applied programming methods, M.V. Yakobovskiy has created algorithms and tools that enable large-scale computational experiments in the field of continuum mechanics on modern and prospective supercomputers with thousands or more processors.

M.V. Yakobovskiy has developed fault-tolerant algorithms for continuous long-term computations on supercomputers with regularly failing nodes and algorithms for guaranteed tetrahedralization of areas defined by triangulated closed surfaces.

M.V. Yakobovskiy participated in the development of Russia's first teraflop supercomputer (RCC RAS), the MVS-15000BM and MVS-100K supercomputers (RCC RAS), the Chebyshev and Lomonosov supercomputers (Moscow State University), and the K-100 heterogeneous supercomputer (Keldysh Institute of Applied Mathematics RAS).

M.V. Yakobovskiy is deeply involved in scientific and organizational work. He is a member of the Scientific Council of the Keldysh Institute of Applied Mathematics RAS, Chairman of Dissertation Council 24.1.237.02, Deputy Chairman of Dissertation Council 24.1.237.01 at the same institute, and a member of Dissertation Council MSU.01.09 at Moscow State University. M.V. Yakobovskiy heads the department of "Software for High-Performance Computing Systems and Networks".

From 2016 to 2019, he was a member of the Presidium of the Higher Attestation Commission under the Ministry of Education and Science of Russia. He is Deputy Editor-in-Chief of the journal "Computational Mathematics and Information Technologies", and a member of the editorial boards of "Supercomputing Frontiers and Innovations", "Computational Methods and Programming", "Advances in Cybernetics", and "Preprints of the Keldysh Institute of Applied Mathematics". He is also a member of the program committees of several international conferences, co-chair of the program committee of the International Congress "Supercomputing Days in Russia", and the All-Russian Conference "Scientific Service on the Internet".

M.V. Yakobovskiy is a member of the Bureau of the Scientific Council of the Russian Academy of Sciences for coordinating scientific research in the field of "Strategic Information Technologies, including the creation of supercomputers and software development", a member of the RAS Scientific Council on Materials and Nanomaterials, a member of the Expert Council of the Russian Foundation for Basic Research (RFBR) on Mathematics and Mechanics, and the scientific secretary of the Scientific Council of the RAS Presidium Program on "Fundamental Principles for Creating Algorithms and Software for Advanced Ultra-High-Performance Computing". He has been an expert for the RAS, RSF, RFBR, and Federal Targeted Programs, and was Chairman of the Expert Council on Directed Basic Research of the RFBR. Under his leadership, several RFBR initiative projects and directed basic research projects have been completed. He has participated

in numerous RSF projects, served as project leader for the RSF, and has been a key executor of several state contracts for Federal Targeted Scientific and Technical Programs and the “Skif-grid” supercomputing program of the Union State, among others.

M.V. Yakobovskiy is heavily involved in educational activities. He is a professor at MIPT and Moscow State University, and the author of four textbooks on parallel algorithms, including the monograph “Introduction to Parallel Methods for Solving Problems”. One of his students was awarded the RAS Medal with a prize for young scientists in 2009 for the work “Modelling Problems of Gas Dynamics and Aeroacoustics Using High-Performance Computing Systems”. He has supervised six candidates of physical and mathematical sciences and more than 30 specialists, bachelors, and masters. In 2016, M.V. Yakobovskiy was elected as a corresponding member of the RAS in the Department of Mathematical Sciences of the RAS, specializing in “Applied Mathematics and Informatics”.

The editorial board of the journal “Computational Mathematics and Information Technologies”, and colleagues of M.V. Yakobovskiy warmly congratulate the esteemed jubilarian, wishing him good health, new ideas, and creative achievements!

Editorial Board

Computational Mathematics and Information Technologies

Boris N. Chetverushkin;
Alexander E. Chistyakov;
Vladimir A. Gasilov;
Valentin A. Gushchin;
Vladimir I. Marchuk;
Alexander P. Ch. Petrov;
Sergey V. Polyakov;
Aleksandr A. Shananin;
Alexander I. Sukhinov;
Vladimir F. Tishkin;
Yuri V. Vasilevsky
Vladimir V. Voevodin.

COMPUTATIONAL MATHEMATICS ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА



Original Theoretical Research

UDC 519.6

<https://doi.org/10.23947/2587-8999-2024-8-2-9-23>


An Adaptive Mesh Refinement Solver for Regularized Shallow Water Equations

Ivan I. But^{1,2} , Maria A. Kiryushina^{1,2} ,
Stepan A. Elistratov^{1,3} , Tatiana G. Elizarova² , Artem D. Tiniakov¹

¹Institute of System Programming of the Russian Academy of Sciences, Moscow, Russian Federation

²Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences, Moscow, Russian Federation

³Shirshov Institute of Oceanology of the Russian Academy of Sciences, Moscow, Russian Federation

m_ist@mail.ru

Abstract

Introduction. We present a novel adaptive mesh refinement (AMR) solver, SWqgdAMR, based on the open software platform AMReX. The new solver is grounded in regularized shallow water equations. This paper details the equations, their discretization, and implementation features within AMReX. The efficacy of SWqgdAMR is demonstrated through two test cases: a two-dimensional circular dam break (collapse of a liquid column) and the collapse of two liquid columns of different heights.

Materials and Methods. The SWqgdAMR solver is developed to extend the applicability of regularized equations in problems requiring high computational power and adaptive grids. SWqgdAMR is the first solver based on the quasigas dynamic (QGD) algorithm within the AMReX framework. The implementation and validation of SWqgdAMR represent a crucial step towards the further expansion of the QGD software suite.

Results. The AMReX-based shallow water equations solver SWqgdAMR with adaptive mesh refinement is described and tested in detail. Validation of SWqgdAMR involved two-dimensional problems: the breach of a cylindrical dam and the breach of two cylindrical dams of different heights. The presented solver demonstrated high efficiency, with the use of adaptive mesh refinement technology accelerating the computation by 56 times compared to a stationary grid calculation.







Discussion and Conclusions. The algorithm can be expanded to include bathymetry, external forces (wind force, bottom friction, Coriolis forces), and the mobility of the shoreline during wetting and drying phases, as has been done in individual codes for regularized shallow water equations (RSWE). The current implementation of the QGD algorithm did not test the potential for parallel computing on graphical cores.

Keywords: shallow water equations, adaptive mesh refinement, quasigas dynamic equations, regularized shallow water equations, AMReX

Funding information. His work was supported by the Moscow Center for Fundamental and Applied Mathematics under Agreement no. 075-15-2022-283 with the Ministry of Science and Higher Education of the Russian Federation.

For citation: But I.I., Kiryushina M.A., Elistratov S.A., Elizarova T.G., Tiniakov A.D. An Adaptive Mesh Refinement Solver for Regularized Shallow Water Equations. *Computational Mathematics and Information Technologies*. 2024;8(2):9–23. <https://doi.org/10.23947/2587-8999-2024-8-2-9-23>

Решатель с адаптивным измельчением сеток для регуляризованных уравнений мелкой воды

И.И. Бут^{1,2} , М.А. Кирюшина^{1,2}  ,
С.А. Елистратов^{1,3} , Т.Г. Елизарова² , А.Д. Тиняков¹ 

¹Институт системного программирования Российской академии наук, Москва, Российская Федерация

²Институт прикладной математики им. М.В. Келдыша Российской академии наук, Москва, Российская Федерация

³Институт океанологии им. П.П. Ширшова Российской академии наук, Москва, Российская Федерация

 m_ist@mail.ru

Аннотация

Введение. Представлен новый решатель с адаптивным измельчением сеток SWqgdAMR на базе открытой программной платформы AMReX. Новый решатель основан на регуляризованных уравнениях мелкой воды. В работе описаны уравнения, их дискретизация и особенности реализации в AMReX. Работоспособность SWqgdAMR была показана на двух тестовых задачах: двумерная задача прорыва круговой дамбы (распад столба жидкости) и задача о распаде двух столбов жидкости, разных по высоте.

Материалы и методы. Решатель SWqgdAMR написан в рамках расширения применимости регуляризованных уравнений в задачах, требующих больших вычислительных мощностей и адаптивных сеток. SWqgdAMR является первым решателем на базе КГД алгоритма в программном комплексе AMReX. Реализация и валидация SWqgdAMR является основным шагом на пути дальнейшего расширения комплекса КГД программ.

Результаты исследования. Детально описан и протестирован решатель AMReX уравнений мелкой воды SWqgdAMR с адаптивным измельчением сеток. Для валидации SWqgdAMR использовались две двумерные задачи: о прорыве цилиндрической плотины и о прорыве двух цилиндрических плотин разной высоты. Представленный решатель показал высокую эффективность, а использование технологии адаптивного измельчения сетки позволило ускорить расчёт в 56 раз по сравнению с расчётом на стационарной сетке.

Обсуждение и заключения. В алгоритм может быть включена батиметрия дна, внешние силы (сила ветра, трение о дно, силы Кориолиса), а также учет подвижности береговой линии при осушении-наводнении, как это уже было сделано в рамках индивидуальных кодов для РУМВ. В данной реализации КГД алгоритма не тестировались перспективные возможности применения распараллеливания вычислений на графические ядра.

Ключевые слова: уравнения мелкой воды, адаптивное измельчение сеток, квазигазодинамические (КГД) уравнения, регуляризованные уравнения мелкой воды (РУМВ), AMReX

Финансирование. Работа выполнена при поддержке Московского центра фундаментальной и прикладной математики. Соглашение с Министерством науки и высшего образования РФ № 075-15-2022-283.

Для цитирования. Бут И.И., Кирюшина М.А., Елистратов С.А., Елизарова Т.Г., Тиняков А.Д. Решатель с адаптивным измельчением сеток для регуляризованных уравнений мелкой воды. *Computational Mathematics and Information Technologies*. 2024;8(2):9–23. <https://doi.org/10.23947/2587-8999-2024-8-2-9-23>

Introduction. Hydro- and gas-dynamics simulations require increasingly precise algorithms and detailed computational grids, which consequently demand substantial computational resources, including methods for parallel computing on GPU cores. Therefore, there is a need to develop a new solver with adaptive mesh refinement (AMR) based on open platforms. This approach offers several advantages over the development of custom codes. Firstly, open platforms typically provide well-established and thoroughly tested frameworks, endorsed by the broader scientific community, reducing the risk of errors and enhancing overall reliability. Secondly, the use of open platforms promotes functional compatibility and reusability, ensuring seamless integration with other tools and facilitating collaboration among researchers. Thirdly, employing existing open platforms can significantly reduce development time and costs, as these platforms often offer a wide range of functionalities, from data processing to visualization and parallel computing. Fourthly, open platforms benefit from continuous development and support from the user community, leading to regular updates, bug fixes, and performance improvements. This contrasts with custom codes, which often depend solely on the resources and expertise of the individual or team that created them.

Among the available open-source software, AMReX was selected as the most optimal framework. AMReX enables:

1. The use of adaptive mesh refinement (AMR) technology.
2. Parallel computation on GPU cores.
3. The immersed boundary method for simulating solid bodies in flow.
4. The construction of structured grids.

5. Integration into The High Performance Software Foundation, established by the Linux Foundation in 2023 [1], ensuring extensive support and ongoing development of this software package.

Numerical simulations of gas dynamics problems have already been conducted using AMReX, including comparisons between AMReX and OpenFOAM [2]. Thus, it was decided to implement a solver for hydro- and gas-dynamics problems based on the quasigas dynamic (QGD) equations within the AMReX framework. A similar solver has already been implemented in OpenFOAM [3, 4] under the general name QGDSolver, demonstrating high efficiency. Unfortunately, as previously noted, OpenFOAM [5] lacks the capabilities for parallel computation on GPU cores and adaptive mesh refinement.

This paper describes the implementation of the QGD algorithm in AMReX in a simplified form. The simplification involves a barotropic variant of the gas dynamics equation system, which allows for the elimination of the energy equation and the equation of state. Under certain assumptions, this barotropic variant takes the form of shallow water equations. It is worth noting that the implementation of the SWqgdAMR solver in the AMReX software package is a key step towards further expanding the suite of solvers based on QGD equations.

The QGD approach itself has been developed for over 30 years for gas dynamics and incompressible flow problems [6–10]. In recent years, the QGD approach has been implemented for shallow water approximation problems [11–16].

Mathematical Model and Numerical Method. Regularized Shallow Water Equations (RSWE). The RSWE can be expressed in vector form, in the absence of external forces and assuming a flat bottom, as follows:

$$\begin{aligned} \frac{\partial h}{\partial t} + \nabla \cdot \mathbf{j}_m &= 0, \\ \frac{\partial(h\mathbf{u})}{\partial t} + \nabla \cdot (\mathbf{j}_m \otimes \mathbf{u}) + \nabla \frac{gh^2}{2} &= \nabla \Pi, \end{aligned} \quad (1)$$

where h is the water layer thickness; $\mathbf{j}_m = h(\mathbf{u} - \mathbf{w})$ is the mass flux density vector; \mathbf{u} is the velocity vector; g is the acceleration due to gravity; $\Pi = \Pi_{NS} + \Pi_{QGD}$ is the stress tensor; Π_{NS} is the Navier-Stokes viscous stress tensor; \mathbf{w} , Π_{QGD} are QGD terms; and \otimes denotes the tensor product. Here, the nabla operator acting on a scalar denotes the gradient, on a vector denotes the divergence, and on a tensor denotes the covariant derivative: $\nabla T \equiv y|_{y_\beta} = \nabla_\alpha T_{\alpha\beta}$. The form of RSWE considering the shape of the bottom and external forces can be found in [11–16].

Discretization of Regularized Shallow Water Equations.

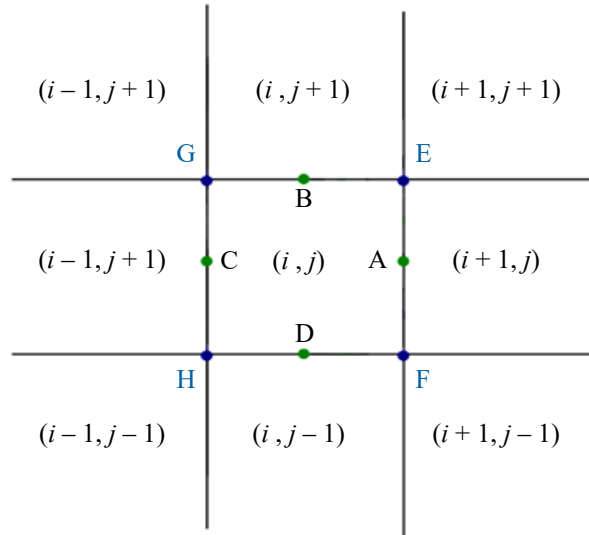


Fig. 1. Numerical stencil. The values of variables h and u are assigned to the cell centers with coordinates (i, j)

The component-wise form of the QGD shallow water equations is as follows:

$$\begin{aligned} \frac{\partial h}{\partial t} + \frac{\partial j_{mx}}{\partial x} + \frac{\partial j_{my}}{\partial y} &= 0, \\ \frac{\partial hu_x}{\partial t} + \frac{\partial j_{mx}u_x}{\partial x} + \frac{\partial j_{my}u_x}{\partial y} + \frac{\partial}{\partial x} \left(\frac{gh^2}{2} \right) &= \frac{\partial \Pi_{xx}}{\partial x} + \frac{\partial \Pi_{yx}}{\partial y}, \\ \frac{\partial hu_y}{\partial t} + \frac{\partial j_{mx}u_y}{\partial x} + \frac{\partial j_{my}u_y}{\partial y} + \frac{\partial}{\partial y} \left(\frac{gh^2}{2} \right) &= \frac{\partial \Pi_{xy}}{\partial x} + \frac{\partial \Pi_{yy}}{\partial y}. \end{aligned}$$

$$\begin{aligned}
 j_{mx} &= h(u_x - w_x), w_x = \frac{\tau}{h} \left(\frac{\partial(hu_x^2)}{\partial x} + \frac{\partial(hu_x u_y)}{\partial y} + gh \frac{\partial h}{\partial x} \right), \\
 j_{my} &= h(u_y - w_y), w_y = \frac{\tau}{h} \left(\frac{\partial(hu_y^2)}{\partial y} + \frac{\partial(hu_x u_y)}{\partial x} + gh \frac{\partial h}{\partial y} \right). \\
 \Pi_{xx} &= u_x \hat{w}_x + R + \Pi_{xx}^{NS}, \Pi_{xy} = u_x \hat{w}_y + \Pi_{xy}^{NS}, \\
 \Pi_{yx} &= u_y \hat{w}_x + \Pi_{yx}^{NS}, \Pi_{yy} = u_y \hat{w}_y + R + \Pi_{yy}^{NS}, \\
 \Pi_{xx}^{NS} &= h\nu \left(2 \frac{\partial u_x}{\partial x} - \frac{2}{3} \text{div} \vec{u} \right), \Pi_{yy}^{NS} = h\nu \left(2 \frac{\partial u_y}{\partial y} - \frac{2}{3} \text{div} \vec{u} \right), \\
 \Pi_{xy}^{NS} &= \Pi_{yx}^{NS} = h\nu \left(\frac{\partial u_y}{\partial x} + \frac{\partial u_x}{\partial y} \right), \nu = \tau \frac{gh^2}{2}, \text{div} \vec{u} = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y}, \\
 \hat{w}_x &= \tau \left(hu_x \frac{\partial u_x}{\partial x} + hu_y \frac{\partial u_x}{\partial y} + \frac{\partial}{\partial x} \left(\frac{gh^2}{2} \right) \right), \\
 \hat{w}_y &= \tau \left(hu_y \frac{\partial u_y}{\partial y} + hu_x \frac{\partial u_y}{\partial x} + \frac{\partial}{\partial y} \left(\frac{gh^2}{2} \right) \right), \\
 R &= g\tau \left(u_x \frac{\partial}{\partial x} \left(\frac{h^2}{2} \right) + u_y \frac{\partial}{\partial y} \left(\frac{h^2}{2} \right) + h^2 \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) \right).
 \end{aligned}$$

To discretize the equations spatially, we take into account the values at the half-cell points of the grid (Fig. 1):

$$\begin{aligned}
 h_A &= h_{i+\frac{1}{2},j} = 0.5(h_{i,j} + h_{i+1,j}), h_B = h_{i,j+\frac{1}{2}} = 0.5(h_{i,j} + h_{i,j+1}), \\
 h_C &= h_{i-\frac{1}{2},j} = 0.5(h_{i,j} + h_{i-1,j}), h_D = h_{i,j-\frac{1}{2}} = 0.5(h_{i,j} + h_{i,j-1}), \\
 h_E &= h_{i+\frac{1}{2},j+\frac{1}{2}} = 0.25(h_{i,j} + h_{i+1,j} + h_{i,j+1} + h_{i+1,j+1}), \\
 h_F &= h_{i+\frac{1}{2},j-\frac{1}{2}} = 0.25(h_{i,j} + h_{i+1,j} + h_{i,j-1} + h_{i+1,j-1}), \\
 h_G &= h_{i-\frac{1}{2},j+\frac{1}{2}} = 0.25(h_{i,j} + h_{i-1,j} + h_{i,j+1} + h_{i-1,j+1}), \\
 h_H &= h_{i-\frac{1}{2},j-\frac{1}{2}} = 0.25(h_{i,j} + h_{i-1,j} + h_{i,j-1} + h_{i-1,j-1}).
 \end{aligned}$$

Similarly, discretization of velocity components at half-cell points is recorded. Discretization of Mass Flux:

$$\begin{aligned}
 j_{xA} &= j_{xi+\frac{1}{2},j} = h_{i+\frac{1}{2},j} \left(u_{xi+\frac{1}{2},j} - w_{xi+\frac{1}{2},j} \right), \\
 j_{yB} &= j_{yi,j+\frac{1}{2}} = h_{i,j+\frac{1}{2}} \left(u_{yi,j+\frac{1}{2}} - w_{yi,j+\frac{1}{2}} \right), \\
 j_{xC} &= j_{xi-\frac{1}{2},j} = h_{i-\frac{1}{2},j} \left(u_{xi-\frac{1}{2},j} - w_{xi-\frac{1}{2},j} \right), \\
 j_{yD} &= j_{yi,j-\frac{1}{2}} = h_{i,j-\frac{1}{2}} \left(u_{yi,j-\frac{1}{2}} - w_{yi,j-\frac{1}{2}} \right).
 \end{aligned}$$

Discretization of QGD Terms:

$$\begin{aligned}
 w_{xA} &= w_{xi+\frac{1}{2},j} = \frac{\tau}{h_{i+\frac{1}{2},j}} \left(\frac{h_{i+1,j} u_{xi+1,j} u_{xi+1,j} - h_{i,j} u_{xi,j} u_{xi,j}}{\Delta x} + \right. \\
 &\quad \left. + \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} - h_{i+\frac{1}{2},j-\frac{1}{2}} u_{xi+\frac{1}{2},j-\frac{1}{2}} u_{xi+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 0.5g \frac{h_{i+1,j} h_{i+1,j} - h_{i,j} h_{i,j}}{\Delta x} \right),
 \end{aligned}$$

$$\begin{aligned}
 w_{yB} &= w_{yi,j+\frac{1}{2}} = \frac{\tau_{i,j+\frac{1}{2}}}{h_{i,j+\frac{1}{2}}} \left(\frac{h_{i,j+1} u_{yi,j+1} u_{yi,j+1} - h_{i,j} u_{yi,j} u_{yi,j}}{\Delta y} + \right. \\
 &+ \left. \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} u_{yi+\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}} u_{yi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} + 0.5g \frac{h_{i,j+1} h_{i,j+1} - h_{i,j} h_{i,j}}{\Delta y} \right), \\
 w_{xC} &= w_{xi-\frac{1}{2},j} = \frac{\tau_{i-\frac{1}{2},j}}{h_{i-\frac{1}{2},j}} \left(\frac{h_{i,j} u_{xi,j} u_{xi,j} - h_{i-1,j} u_{xi-1,j} u_{xi-1,j}}{\Delta x} + \right. \\
 &+ \left. \frac{h_{i-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}} u_{yi-\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}} u_{xi-\frac{1}{2},j-\frac{1}{2}} u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 0.5g \frac{h_{i,j} h_{i,j} - h_{i-1,j} h_{i-1,j}}{\Delta x} \right), \\
 w_{yD} &= w_{yi,j-\frac{1}{2}} = \frac{\tau_{i,j-\frac{1}{2}}}{h_{i,j-\frac{1}{2}}} \left(\frac{h_{i,j} u_{yi,j} u_{yi,j} - h_{i,j-1} u_{yi,j-1} u_{yi,j-1}}{\Delta y} + \right. \\
 &+ \left. \frac{h_{i+\frac{1}{2},j-\frac{1}{2}} u_{xi+\frac{1}{2},j-\frac{1}{2}} u_{yi+\frac{1}{2},j-\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}} u_{xi-\frac{1}{2},j-\frac{1}{2}} u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} + 0.5g \frac{h_{i,j} h_{i,j} - h_{i,j-1} h_{i,j-1}}{\Delta y} \right), \\
 w_{xA}^* &= w_{xi+\frac{1}{2},j}^* = \tau_{i+\frac{1}{2},j} \left(h_{i+\frac{1}{2},j} u_{xi+\frac{1}{2},j} \frac{u_{xi+1,j} - u_{xi,j}}{\Delta x} + h_{i+\frac{1}{2},j} u_{yi+\frac{1}{2},j} \frac{u_{xi+\frac{1}{2},j+\frac{1}{2}} - u_{xi+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + \right. \\
 &+ \left. 0.5g \frac{h_{i+1,j} h_{i+1,j} - h_{i,j} h_{i,j}}{\Delta x} \right), \\
 w_{xB}^* &= w_{xi,j+\frac{1}{2}}^* = \tau_{i,j+\frac{1}{2}} \left(h_{i,j+\frac{1}{2}} u_{xi,j+\frac{1}{2}} \frac{u_{xi+\frac{1}{2},j+\frac{1}{2}} - u_{xi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} + h_{i,j+\frac{1}{2}} u_{yi,j+\frac{1}{2}} \frac{u_{xi,j+1} - u_{xi,j}}{\Delta y} + \right. \\
 &+ \left. 0.5g \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} h_{i+\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j+\frac{1}{2}} h_{i-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} \right), \\
 w_{xC}^* &= w_{xi-\frac{1}{2},j}^* = \tau_{i-\frac{1}{2},j} \left(h_{i-\frac{1}{2},j} u_{xi-\frac{1}{2},j} \frac{u_{xi,j} - u_{xi-1,j}}{\Delta x} + h_{i-\frac{1}{2},j} u_{yi-\frac{1}{2},j} \frac{u_{xi-\frac{1}{2},j+\frac{1}{2}} - u_{xi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + \right. \\
 &+ \left. 0.5g \frac{h_{i,j} h_{i,j} - h_{i-1,j} h_{i-1,j}}{\Delta x} \right), \\
 w_{xD}^* &= w_{xi,j-\frac{1}{2}}^* = \tau_{i,j-\frac{1}{2}} \left(h_{i,j-\frac{1}{2}} u_{xi,j-\frac{1}{2}} \frac{u_{xi+\frac{1}{2},j-\frac{1}{2}} - u_{xi-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} + h_{i,j-\frac{1}{2}} u_{yi,j-\frac{1}{2}} \frac{u_{xi,j} - u_{xi,j-1}}{\Delta y} + \right. \\
 &+ \left. 0.5g \frac{h_{i+\frac{1}{2},j-\frac{1}{2}} h_{i+\frac{1}{2},j-\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}} h_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} \right), \\
 w_{yA}^* &= w_{yi+\frac{1}{2},j}^* = \tau_{i+\frac{1}{2},j} \left(h_{i+\frac{1}{2},j} u_{yi+\frac{1}{2},j} \frac{u_{yi+1,j} - u_{yi,j}}{\Delta x} + h_{i+\frac{1}{2},j} u_{xi+\frac{1}{2},j} \frac{u_{yi+\frac{1}{2},j+\frac{1}{2}} - u_{yi+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + \right. \\
 &+ \left. 0.5g \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} h_{i+\frac{1}{2},j+\frac{1}{2}} - h_{i+\frac{1}{2},j-\frac{1}{2}} h_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} \right),
 \end{aligned}$$

$$\begin{aligned}
 w_{yB}^* &= w_{yi,j+\frac{1}{2}}^* = \tau_{i,j+\frac{1}{2}} \left(h_{i,j+\frac{1}{2}} u_{xi,j+\frac{1}{2}} \frac{u_{yi+\frac{1}{2},j+\frac{1}{2}} - u_{yi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} + h_{i,j+\frac{1}{2}} u_{yi,j+\frac{1}{2}} \frac{u_{yi,j+1} - u_{yi,j}}{\Delta y} + \right. \\
 &\quad \left. + 0.5g \frac{h_{i,j+1}h_{i,j+1} - h_{i,j}h_{i,j}}{\Delta y} \right), \\
 w_{yC}^* &= w_{yi-\frac{1}{2},j}^* = \tau_{i-\frac{1}{2},j} \left(h_{i-\frac{1}{2},j} u_{xi-\frac{1}{2},j} \frac{u_{yi,j} - u_{yi-1,j}}{\Delta x} + h_{i-\frac{1}{2},j} u_{yi-\frac{1}{2},j} \frac{u_{yi-\frac{1}{2},j+\frac{1}{2}} - u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + \right. \\
 &\quad \left. + 0.5g \frac{h_{i-\frac{1}{2},j+\frac{1}{2}}h_{i-\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}}h_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} \right), \\
 w_{yD}^* &= w_{yi,j-\frac{1}{2}}^* = \tau_{i,j-\frac{1}{2}} \left(h_{i,j-\frac{1}{2}} u_{xi,j-\frac{1}{2}} \frac{u_{yi+\frac{1}{2},j-\frac{1}{2}} - u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} + h_{i,j-\frac{1}{2}} u_{yi,j-\frac{1}{2}} \frac{u_{yi,j} - u_{yi,j-1}}{\Delta y} + \right. \\
 &\quad \left. + 0.5g \frac{h_{i,j}h_{i,j} - h_{i,j-1}h_{i,j-1}}{\Delta y} \right), \\
 \text{div} \vec{u}_A &= \frac{u_{xi+1,j} - u_{xi,j}}{\Delta x} + \frac{u_{yi+\frac{1}{2},j+\frac{1}{2}} - u_{yi-\frac{1}{2},j+\frac{1}{2}}}{\Delta y}, \text{div} \vec{u}_B = \frac{u_{yi,j+1} - u_{yi,j}}{\Delta y} + \frac{u_{xi+\frac{1}{2},j+\frac{1}{2}} - u_{xi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x}, \\
 \text{div} \vec{u}_C &= \frac{u_{xi,j} - u_{xi-1,j}}{\Delta x} + \frac{u_{yi-\frac{1}{2},j+\frac{1}{2}} - u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y}, \text{div} \vec{u}_D = \frac{u_{yi,j} - u_{yi,j-1}}{\Delta y} + \frac{u_{xi+\frac{1}{2},j-\frac{1}{2}} - u_{xi-\frac{1}{2},j-\frac{1}{2}}}{\Delta x}.
 \end{aligned}$$

The regularization parameter of the algorithm

$$\tau = \frac{\alpha \Delta_h}{\sqrt{gh}} \quad (2)$$

is calculated as

$$\tau_A = \tau_{i+\frac{1}{2},j} = \alpha \sqrt{\frac{\Delta x^2 + \Delta y^2}{gh_{i+\frac{1}{2},j}}}, \quad \tau_B = \tau_{i,j+\frac{1}{2}} = \alpha \sqrt{\frac{\Delta x^2 + \Delta y^2}{gh_{i,j+\frac{1}{2}}}}, \quad (3)$$

where α is a tuning parameter between 0 and 1; g is the acceleration due to gravity. A similar discretization is applied for terms τ_C, τ_D . The time step on the base computational grid is chosen to satisfy the stability condition for the explicit scheme, expressed as the Courant condition (Courant number $0 < \beta < 1$):

$$\Delta t = \beta \left(\frac{\Delta x + \Delta y}{2\sqrt{gh}} \right)_{\min}, \quad (4)$$

$$\begin{aligned}
 R_A &= R_{i+\frac{1}{2},j} = \frac{g\tau_{i+\frac{1}{2},j}}{2} \left(u_{xi+\frac{1}{2},j} \frac{h_{i+1,j}h_{i+1,j} - h_{i,j}h_{i,j}}{\Delta x} + \right. \\
 &\quad \left. + u_{yi+\frac{1}{2},j} \frac{h_{i+\frac{1}{2},j+\frac{1}{2}}h_{i+\frac{1}{2},j+\frac{1}{2}} - h_{i+\frac{1}{2},j-\frac{1}{2}}h_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 2h_{i+\frac{1}{2},j}h_{i+\frac{1}{2},j} \text{div}(u)_A \right), \\
 R_B &= R_{i,j+\frac{1}{2}} = \frac{g\tau_{i,j+\frac{1}{2}}}{2} \left(u_{yi,j+\frac{1}{2}} \frac{h_{i,j+1}h_{i,j+1} - h_{i,j}h_{i,j}}{\Delta y} + \right. \\
 &\quad \left. + u_{xi,j+\frac{1}{2}} \frac{h_{i-\frac{1}{2},j+\frac{1}{2}}h_{i-\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}}h_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} + 2h_{i,j+\frac{1}{2}}h_{i,j+\frac{1}{2}} \text{div}(u)_B \right),
 \end{aligned}$$

$$\begin{aligned}
 R_C = R_{i-\frac{1}{2},j} &= \frac{g\tau_{i-\frac{1}{2},j}}{2} \left(u_{xi-\frac{1}{2},j} \frac{h_{i,j}h_{i,j} - h_{i-1,j}h_{i-1,j}}{\Delta x} + \right. \\
 &\quad \left. + u_{yi-\frac{1}{2},j} \frac{h_{i-\frac{1}{2},j+\frac{1}{2}}h_{i-\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}}h_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 2h_{i-\frac{1}{2},j}h_{i-\frac{1}{2},j} \operatorname{div}(u)_C \right), \\
 R_D = R_{i,j-\frac{1}{2}} &= \frac{g\tau_{i,j-\frac{1}{2}}}{2} \left(u_{yi,j-\frac{1}{2}} \frac{h_{i,j}h_{i,j} - h_{i,j-1}h_{i,j-1}}{\Delta y} + \right. \\
 &\quad \left. + u_{xi,j-\frac{1}{2}} \frac{h_{i+\frac{1}{2},j-\frac{1}{2}}h_{i+\frac{1}{2},j-\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}}h_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta x} + 2h_{i,j-\frac{1}{2}}h_{i,j-\frac{1}{2}} \operatorname{div}(u)_D \right).
 \end{aligned}$$

Discretization of the viscous stress tensor:

$$\begin{aligned}
 \Pi_{xxA} = \Pi_{xxi+\frac{1}{2},j} &= u_{xi+\frac{1}{2},j} w_{xi+\frac{1}{2},j}^* + R_{xi+\frac{1}{2},j}, \quad \Pi_{xxB} = \Pi_{xxi-\frac{1}{2},j} = u_{xi-\frac{1}{2},j} w_{xi-\frac{1}{2},j}^* + R_{xi-\frac{1}{2},j}, \\
 \Pi_{xyA} = \Pi_{xyi+\frac{1}{2},j} &= u_{xi+\frac{1}{2},j} w_{yi+\frac{1}{2},j}^*, \quad \Pi_{xyB} = \Pi_{xyi-\frac{1}{2},j} = u_{xi-\frac{1}{2},j} w_{yi-\frac{1}{2},j}^*, \\
 \Pi_{yxC} = \Pi_{yxi,j+\frac{1}{2}} &= u_{yi,j+\frac{1}{2}} w_{xi,j+\frac{1}{2}}^*, \quad \Pi_{yxD} = \Pi_{yxi,j-\frac{1}{2}} = u_{yi,j-\frac{1}{2}} w_{xi,j-\frac{1}{2}}^*, \\
 \Pi_{yyC} = \Pi_{yyi,j+\frac{1}{2}} &= u_{xi,j+\frac{1}{2}} w_{xi,j+\frac{1}{2}}^* + R_{xi,j+\frac{1}{2}}, \quad \Pi_{yyD} = \Pi_{yyi,j-\frac{1}{2}} = u_{xi,j-\frac{1}{2}} w_{xi,j-\frac{1}{2}}^* + R_{xi,j-\frac{1}{2}}.
 \end{aligned}$$

Discretization of the mass conservation equation:

$$\hat{h}_{i,j} = h_{i,j} - \frac{\Delta t}{\Delta x} \left((j_x)_{i+\frac{1}{2},j} - (j_x)_{i-\frac{1}{2},j} \right) - \frac{\Delta t}{\Delta y} \left((j_y)_{i,j+\frac{1}{2}} - (j_y)_{i,j-\frac{1}{2}} \right).$$

Discretization of the momentum balance equations:

$$\begin{aligned}
 \hat{h}_{i,j}(\hat{u}_x)_{i,j} &- \frac{\Delta t}{\Delta x} \left((u_x)_{i+\frac{1}{2},j} (j_x)_{i+\frac{1}{2},j} - (u_x)_{i-\frac{1}{2},j} (j_x)_{i-\frac{1}{2},j} \right) - \frac{\Delta t}{\Delta y} \left((u_x)_{i,j+\frac{1}{2}} (j_y)_{i,j+\frac{1}{2}} - (u_x)_{i,j-\frac{1}{2}} (j_y)_{i,j-\frac{1}{2}} \right) - \\
 &- 0.5g \frac{\Delta t}{\Delta x} \left(h_{i+\frac{1}{2},j} h_{i+\frac{1}{2},j} - h_{i-\frac{1}{2},j} h_{i-\frac{1}{2},j} \right) + \frac{\Delta t}{\Delta x} \left((\Pi_{xx})_{i+\frac{1}{2},j} - (\Pi_{xx})_{i-\frac{1}{2},j} \right) + \frac{\Delta t}{\Delta y} \left((\Pi_{yx})_{i,j+\frac{1}{2}} - (\Pi_{yx})_{i,j-\frac{1}{2}} \right), \\
 \hat{h}_{i,j}(\hat{u}_y)_{i,j} &= h_{i,j}(u_y)_{i,j} - \frac{\Delta t}{\Delta y} \left((u_y)_{i,j+\frac{1}{2}} (j_y)_{i,j+\frac{1}{2}} - (u_y)_{i,j-\frac{1}{2}} (j_y)_{i,j-\frac{1}{2}} \right) - \\
 &- \frac{\Delta t}{\Delta x} \left((u_y)_{i+\frac{1}{2},j} (j_x)_{i+\frac{1}{2},j} - (u_y)_{i-\frac{1}{2},j} (j_x)_{i-\frac{1}{2},j} \right) - 0.5g \frac{\Delta t}{\Delta y} \left(h_{i,j+\frac{1}{2}} h_{i,j+\frac{1}{2}} - h_{i,j-\frac{1}{2}} h_{i,j-\frac{1}{2}} \right) + \\
 &+ \frac{\Delta t}{\Delta x} \left((\Pi_{yy})_{i,j+\frac{1}{2}} - (\Pi_{yy})_{i,j-\frac{1}{2}} \right) + \frac{\Delta t}{\Delta x} \left((\Pi_{xy})_{i+\frac{1}{2},j} - (\Pi_{xy})_{i-\frac{1}{2},j} \right).
 \end{aligned}$$

Implementation in AMReX. The numerical solution of the shallow water equations is implemented in C++ using the open-source software AMReX. This software was chosen as the foundation because it facilitates ready-to-use adaptive mesh refinement (AMR) logic and offers straightforward portability of computations to GPU cores via macros, significantly reducing computational time.

Figure 2 shows the structure of the developed software.

The main solver class, AmrSWQGD, is declared in the file AmrSWQGD.H and implemented in the file AmrSWQGD.cpp. It inherits from the AmrLevel class, defined in the AMReX core. Inheriting from this class allows straightforward adaptive mesh refinement across multiple levels.

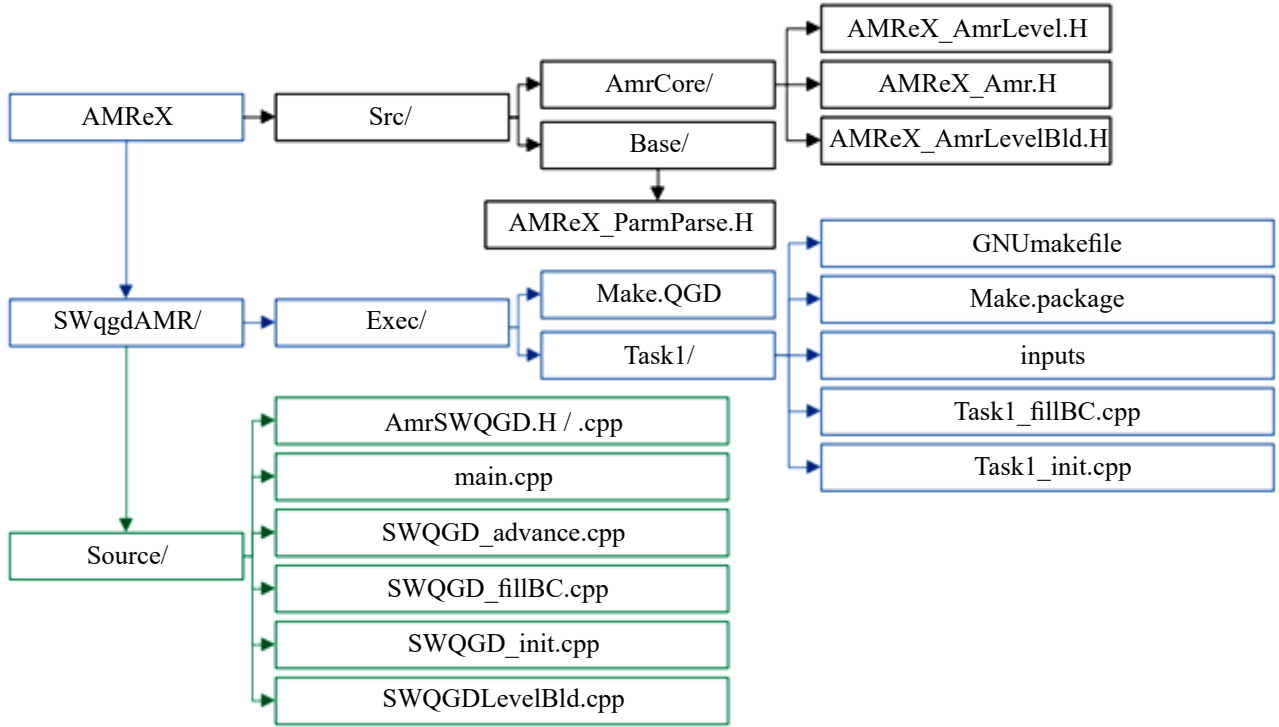


Fig. 2. Solver structure

The solver iteration logic is defined in the file `SWQGD_advance.cpp` within the `advance` method. In this method, a `ParallelFor` loop updates the fields `h`, `ux` and `uy` at each iteration. Here, `huOld` is the container for old variables, and `huNew` is for new variables. Since the solution occurs on a multi-level grid (Fig. 3), where each level has its own refinement (level 0 is the coarsest grid, and higher levels have increased accuracy), each level introduces its own time step (Fig. 4).

For example, if the grid has two levels, 0 and 1, and the grid at level 1 is twice as fine in each direction as at level 0, then one iteration of the solution proceeds as follows: calculations are performed at level 0 with a time step dt , two iterations of calculations are performed at level 1 with a time step $dt/2$, and then the grids are synchronized. This algorithm enhances computational accuracy.

It is important to note that it is not necessary to refine the entire grid at each level, only specific parts of it. To achieve this, the solver class defines the `errorEst` method in the file `AmrSWQGD.cpp`. This method takes a reference to an instance of the `TagBoxArray` container. Using a `ParallelFor` loop, each grid cell is examined and marked for refinement if it meets certain conditions (defined within an `if` statement). Additionally, some surrounding cells are marked for refinement. Cells that do not meet the condition are marked with the `clearval` tag and will not be refined.

The computational tasks themselves are located in the `Exec` directory, which contains the `inputs` files with initial and boundary conditions.

The `inputs` file contains the settings for the solution, including various parameters that control the behavior of the solver. Here are the key parameters and their descriptions:

- `max_step`: The maximum number of iterations;
- `stop_time`: The computational time in seconds at which the solution stops. Essentially, the calculations continue until either the number of iterations exceeds `max_step` or the computational time reaches `stop_time`;
- `geometry.is_periodic`: An array of three boolean variables (e.g., 0 0 0, 0 1 1, or 1 0 1) that determine whether the boundaries in each direction are periodic (1) or not (0);
- `geometry.coord_sys`: The coordinate system used for the solution. The recommended value is 0, which corresponds to the Cartesian coordinate system. There is no guarantee that the solver will work correctly in other coordinate systems;
- `geometry.prob_lo`: The xyz coordinates of the lower left corner of the physical rectangular domain (e.g., 0.0 0.0 0.0);
- `geometry.prob_hi`: The xyz coordinates of the upper right corner of the physical domain (e.g., 10.0 10.0 1.0);
- `amr.n_cell`: An array of three integers representing the grid resolution in each direction at level 0 (e.g., 512 512 1);
- `amr.max_level`: An integer indicating the maximum allowable level of grid refinement;
- `amr.ref_ratio`: The refinement ratio of the grid levels;
- `amr.regrid_int`: An integer representing the number of steps after which the grid should be regenerated;
- `amr.max_grid_size`: The solver divides the grid into domains, with the size of each domain not exceeding `max_grid_size` in each direction;

- `amr.plot_int`: The solver writes output to files every `plot_int` iterations.

The boundary conditions are defined in the structure `SWQGDBCFill` located in the file `Task1_fillBC.cpp` (where `Task1` is the task name and can be changed to any other name). In this structure: `ilo` and `ihi` represent the left and right boundaries of the computational domain, respectively, `jlo` and `jhi` represent the bottom and top boundaries, respectively. The equations are solved for the variables h , u_x and u_y , which represent the water column height, the velocity in the x -direction, and the velocity in the y -direction, respectively. The variable `dest` is a multidimensional array that stores the values of all variables throughout the computational domain. To access the variable h at cell (i, j, k) you use `dest(i, j, k, 0)`. To access the variable u_x in the same cell, you use `dest(i, j, k, 1)`, and for u_y , you use `dest(i, j, k, 2)`. Since we are dealing with 2D equations, the index k is assumed to be 1.

Initial conditions are set in the file `Task1_init.cpp`. Here, the variable `snew[bi]` acts as a container similar to the `dest` container in the boundary conditions structure. Initial conditions are computed in a loop using `amrex::ParallelFor`. This loop, along with the macro `AMREX_GPU_DEVICE`, allows for the parallel computation of initial conditions across the entire domain on GPU cores.

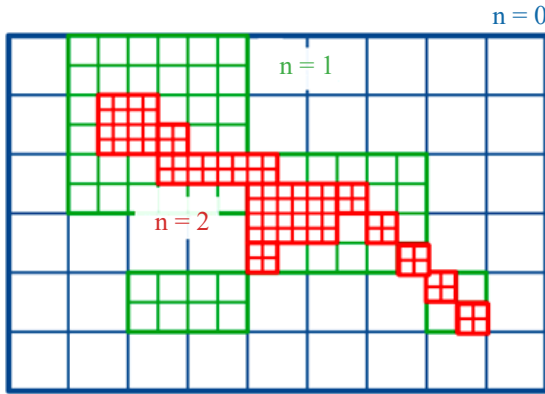


Fig. 3. Adaptive Mesh Refinement Algorithm

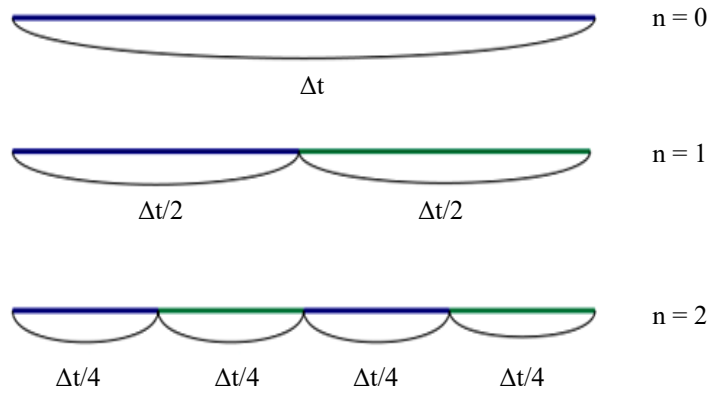


Fig. 4. Time Step Splitting Algorithm in Subcycle

To compile the program, navigate to the task folder (`Task1`) and then run the command `make -j n` in the terminal, where n is the number of cores for parallelization. To run the utility without parallelization, simply run `make`. After `make` completes, a file named `main2d.gnu.MPI.ex` (the name may vary slightly) will appear in the folder. To start the calculation, execute the following command in the terminal: `mpirun -np n ./main2d.gnu.MPI.ex inputs`, where n is the number of cores for parallelization. This will start the computation.

Results. Two 2D problems are used for the validation and verification of the developed solver:

The dam break problem, for which an analytical solution is well known.

The collapse of two liquid columns of different heights.

Two-Dimensional Circular Dam-Break Problem. The problem of liquid column collapse or the breakthrough of a circular dam (Circular Dam-Break) is widely used in the validation and verification of new solvers [17–20]. Consider a 2D plane with dimensions 40×40 m, where at the center resides a liquid column with height $h = 2.5$ m and radius $R = 2.5$ m. The height of the liquid in the rest of the domain is $h_0 = 0.5$ m (see Fig. 5). The computational domain is divided into 40,000 uniform cells, i. e., 200 cells in each direction. The time step is chosen as $\Delta t = 10^{-4}$ s, and the calculation is carried out until time $t = 4.7$ s.

Visualization in Figure 6 illustrates the liquid column collapse. Initially, the wall is removed, allowing the water to move in all directions. As the circular shock wave propagates outward, a rarefaction wave moves inward into the original cylinder until it completely converges at the center of the computational domain, where it reflects, causing a height gradient and hence a secondary shock wave. Results of numerical experiments compared with the analytical solution from [19] at time $t = 4.7$ s are shown in Fig. 7. Panel (a) of Fig. 7 demonstrates the dependence of the solution on the algorithm tuning parameter α . The optimal value is $\alpha = 0.2$. Panel (b) of Figure 7 illustrates the convergence of the solution with grid refinement. The characteristic Courant number is 0.2.

An example of the adaptive mesh refinement algorithm can be seen in Figure 8. Depending on the chosen adaptation criterion (in our case, the gradient of the water column height), the mesh is refined across levels (in our case $n_{\text{amr}} = 4$, where n_{amr} is the maximum level in the current calculation), significantly accelerating the computation. Detailed investigation is presented in the section on performance evaluation of the `SWqgdAMR` solver.

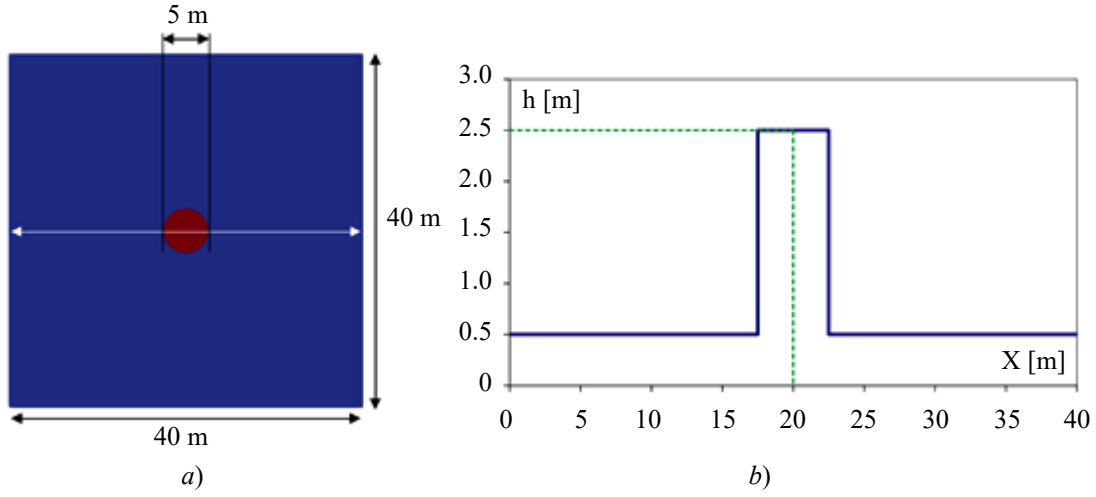


Fig. 5. Initial Conditions of the Circular Dam-Break Problem: *a* — Computational domain geometry and initial distribution; *b* — Height of the liquid column along the white line

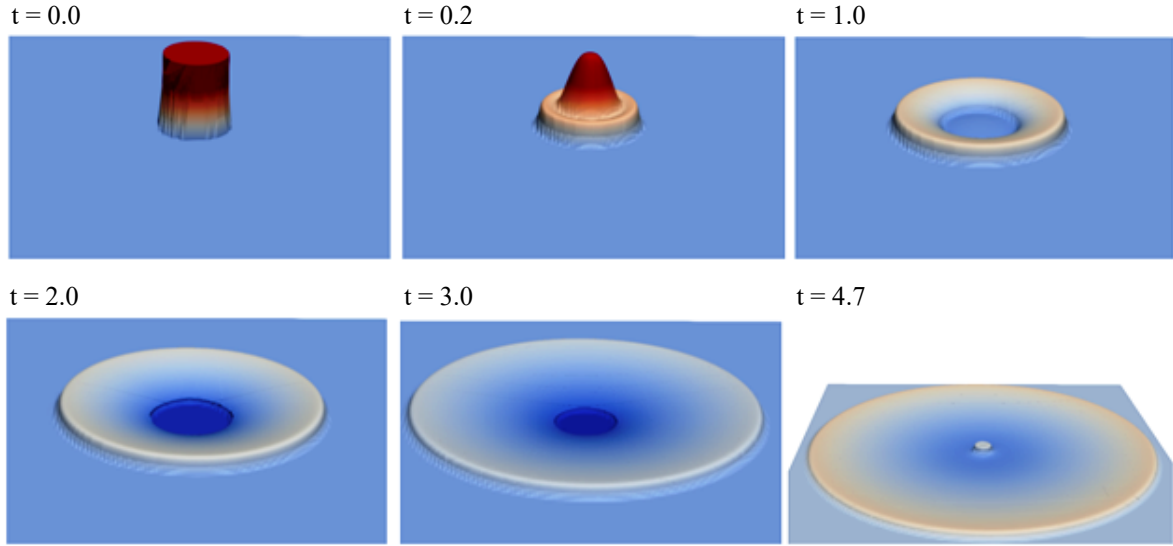


Fig. 6. Visualization of Liquid Column Collapse Over Time $\alpha = 0.2$, $\Delta t = 10^{-4}$ s.
Time in seconds on the figure

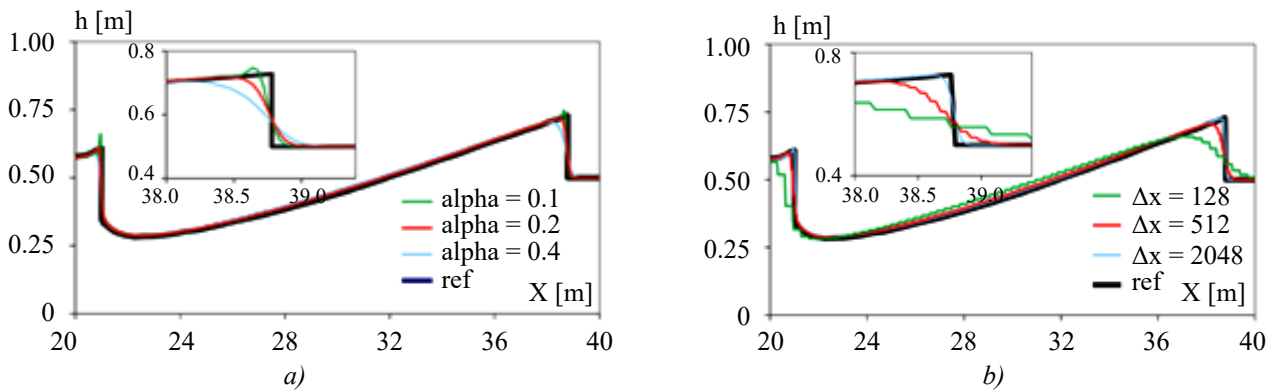


Fig. 7. Distribution of Liquid Column Height with Time Step $\Delta t = 10^{-4}$ s at $t = 4.7$ s:
a — Dependence on parameter α on a fixed grid $\Delta x = 1024$;
b — Grid convergence with constant $\alpha = 0.2$

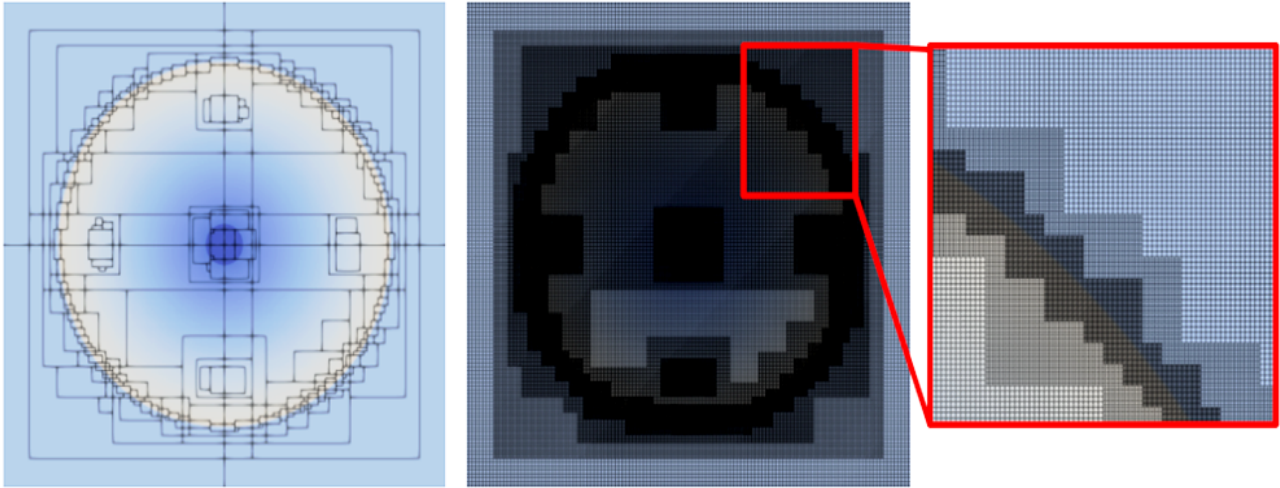


Fig. 8. Visualization of Adaptive Mesh Refinement Algorithm at $\alpha = 0.2$ at time $t = 3$ s. The first image shows the block decomposition, while the second image depicts the adaptive mesh

The collapse of two different-height liquid columns problem allows the solver's capability to reproduce complex flow structures to be tested, similar to the previous test. Considered is a 2D plane of size 2000×2000 m, with the first water column located at (875.0) having a radius $R_1 = 125$ m and height $h_1 = 15$ m. At coordinates (1375.0) the second water column has radius $R_2 = 125$ m, $h_2 = 20$ m, with the water level in the remaining area set to $h_1 = 10$ m (see Fig. 9). The computational domain is divided into 160,000 uniform cells, i. e., 400 cells in each direction. The time step is chosen as $\Delta t = 10^{-4}$ s, and the computation concludes at time $t = 30$ s.

Fig. 10 and 11 visualize the collapse and subsequent interaction of the two liquid columns. Initially, the walls are removed, allowing water to move in all directions from each column. Subsequently, two shock waves collide, resulting in significant deformation of the wave fronts.

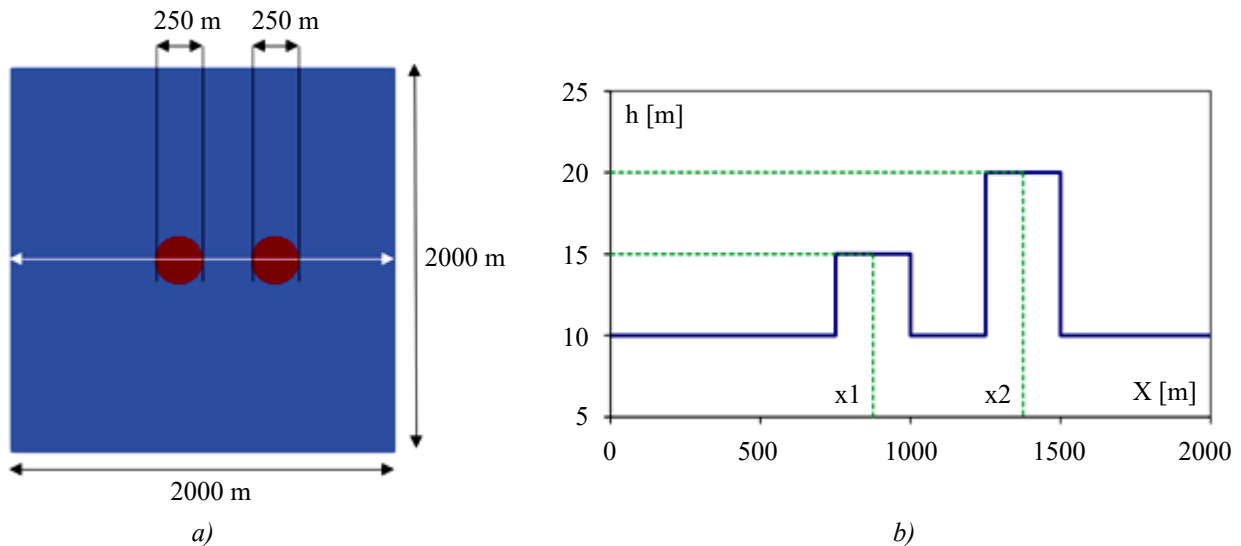


Fig. 9. Initial Conditions of the Liquid Column Collapse Problem: *a* — geometry of the computational domain; *b* — height of the liquid column along the white line, $x_1 = 875$ m, $x_2 = 1375$ m

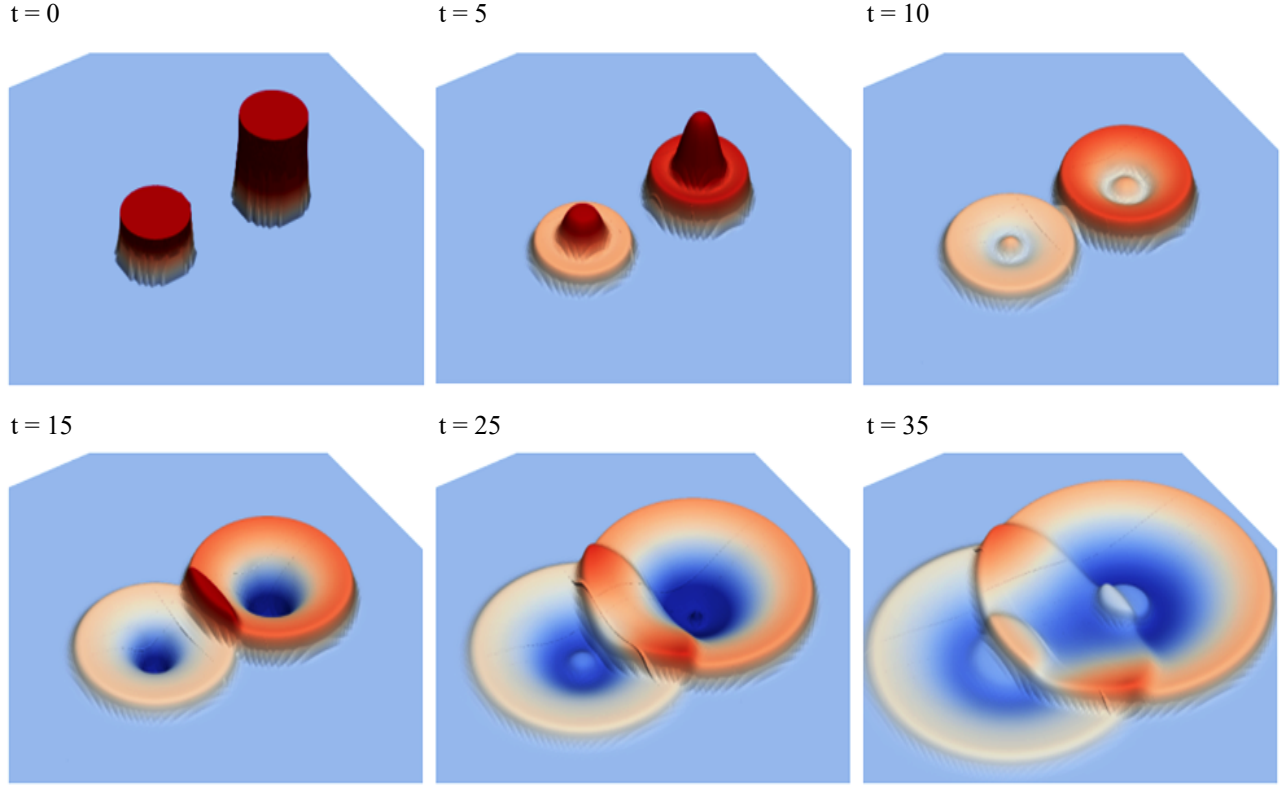


Fig. 10. Visualization of the collapse of two liquid columns over time $\alpha = 0.2$, $\Delta t = 10^{-4}$ s.
Time in the figure is in seconds

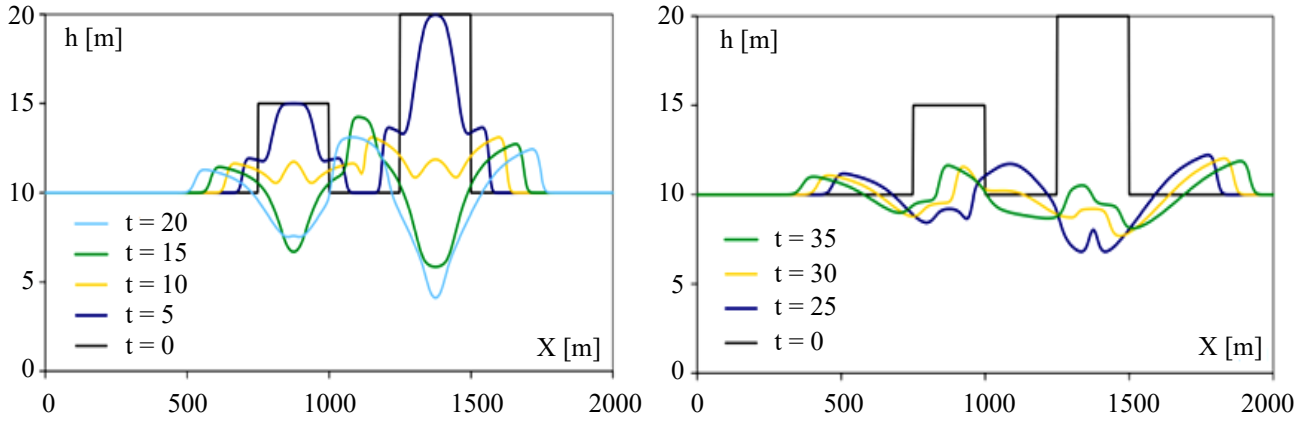


Fig. 11. Plot of the collapse of two liquid columns over time in the central cross-section.
Time in the figure is in seconds, $\Delta x = 1024$, $\alpha = 0.2$, $\Delta t = 10^{-4}$ s

Performance of the SWqgdAMR Solver. One of the crucial criteria in developing a new solver is assessing its performance and the efficiency of parallelization. For this purpose, the problem of the collapse of two different-height liquid columns was utilized. A computational grid of 1.048.576 cells, a time step of $\Delta t = 10^{-4}$ s, and computation completion at $t = 0.1$ c. Performance evaluation was conducted on an Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz with 8 cores, and the results are presented in Table 1.

Table 1

Performance Evaluation of the SWqgdAMR Solver

Number of Cores	Number of Cells	Computation Time t , s	Efficiency, %
1	1 048 576	71	—
2	524 288	40	89
4	262 144	27	66

Using the same processor with 2 cores, acceleration of computations was investigated through the use of adaptive mesh refinement technology. The data are presented in Table 2. The number of computational cells at $n_{\text{amr}} = 0$ is 4 194 304, with a base grid at $n_{\text{amr}} = 1$ of 1 048 576, at $n_{\text{amr}} = 2$ of 262 144, at $n_{\text{amr}} = 4$ base computational grid of 65 536. In all cases, flow resolution remains constant, with quality varying depending on the adaptation criterion (an example of mesh refinement algorithm operation is shown in Figure 8). In our calculations, the gradient of the liquid column height was chosen as the mesh adaptation criterion.

Table 2

Computation time in seconds depending on levels of mesh adaptation

Number of Cores	Number of Cells	$n_{\text{amr}} = 0$	$n_{\text{amr}} = 1$	$n_{\text{amr}} = 2$	$n_{\text{amr}} = 4$
2	4 194 304	2288	359	132	41

Thus, the use of adaptive mesh refinement technology allows for significant acceleration of computations. In our case, acceleration of up to 56 times was achieved. In [21], it was demonstrated that on identical stationary grids, AMReX is 4 times faster than OpenFOAM, indicating that with mesh refinement, speed gains of up to 232 times can be achieved.

Discussion and Conclusions. The AMReX solver for shallow water equations (SWqgdAMR) with adaptive mesh refinement (AMR) was comprehensively described and tested in this work. Two 2D test cases were used for validation: the breach of a cylindrical dam and the breach of two cylindrical dams of different heights. The presented solver demonstrated high efficiency, and the use of adaptive mesh refinement technology accelerated computations by a factor of 56 compared to computations on a stationary grid.

The SWqgdAMR solver was developed as part of efforts to expand the applicability of regularized equations in problems requiring significant computational resources and adaptive grids. It represents the first solver based on the shallow water equations algorithm within the AMReX software framework. The implementation and validation of SWqgdAMR represent a crucial step towards further expanding the suite of shallow water equations programs. Future work will include incorporating quasi-gasdynamic equations into AMReX for simulating gas dynamics problems.

In this implementation, the prospective capabilities of leveraging graphics processing unit (GPU) computing for parallel computation were not tested. Additionally, it is noteworthy that the algorithm could be extended to include bathymetry, external forces (such as wind force, bottom friction, and Coriolis forces), and consideration of shoreline mobility during flooding and drying, as has been implemented in individual codes for hydrodynamic simulations.

References

1. TheLinuxFoundation. Linux Foundation Announces Intent to Form the High Performance Software Foundation. URL: <https://www.linuxfoundation.org/press/linux-foundation-announces-intent-to-form-high-performance-software-foundation-hpsf> (accessed: 16.04.2024).
2. Epikhin A., But I. Numerical Simulation of Supersonic Jet Noise Using Open Source Software. *International Conference on Computational Science*. Springer. 2023. P. 292–302.
3. Kraposhin M.V. et al. Development of a new OpenFOAM solver using regularized gas dynamic equations. *Computers & Fluids*. 2018;166:163–175.
4. Kraposhin M.V., Ryazanov D.A., Elizarova T.G. Numerical algorithm based on regularized equations for incompressible flow modeling and its implementation in OpenFOAM. *Computer Physics Communications*. 2022;271: 108216.
5. QGDsolver. URL: <https://github.com/unicfdlab/QGDsolver> (accessed: 16.04.2024).
6. Elizarova T.G. *Quasi-gas-dynamic Equations*. Springer. 2009.
7. Elizarova T.G. *Quasi-gas-dynamic Equations and Methods for Calculating Viscous Flows*. Moscow: Nauchnyi Mir; 2007. 350 p. (in Russ.).
8. Chetverushkin B.N. *Kinetic Schemes and Quasi-gas-dynamic System of Equations*. Moscow: MAKSPress; 2004. 328 p. (in Russ.).
9. Sheretov Yu.V. *Regularized Equations of Hydrodynamics*. Tver: Tver State University; 2016. 222 p. (in Russ.).
10. Sheretov Yu.V. *Dynamics of Continua under Space-Time Averaging*. Moscow; Izhevsk: Regular and Chaotic Dynamics; 2009. 400 p. (in Russ.).
11. Bulatov O., Elizarova T.G. Regularized shallow water equations and an efficient method for numerical simulation of shallow water flows. *Computational mathematics and mathematical physics*. 2011;51:160–173.
12. Elizarova T.G., Ivanov A.V. Regularized equations for numerical simulation of flows in the two-layer shallow water approximation. *Computational Mathematics and Mathematical Physics*. 2018;58:714–734.
13. Saburin D.S., Elizarova T.G. Modelling the Azov Sea circulation and extreme surges in 2013–2014 using the regularized shallow water equations. *Russian Journal of Numerical Analysis and Mathematical Modelling*. 2018;33(3):173–185.

14. Bulatov O.V., Elizarova T.G. Regularized Shallow Water Equations and an Efficient Method for Numerical Simulation of Flows in Shallow Water Bodies. *Journal of Computational Mathematics and Mathematical Physics*. 2011;51(1):170–184. (in Russ.).
15. Elizarova T.G., Ivanov A.V. Regularized Equations for Numerical Simulation of Flows in the Two-Layer Shallow Water Approximation. *Journal of Computational Mathematics and Mathematical Physics*. 2018;58(5):741–761. (in Russ.).
16. Ivanov A.V. On the Implementation of the Shallow Water Model Based on the Quasi-gas-dynamic Approach in the OpenFOAM Open Source Software. *Preprints of the Institute of Applied Mathematics named after M.V. Keldysh RAS*. 2023;28:27. (in Russ.).
17. Delis A., Nikolos I. A novel multidimensional solution reconstruction and edge-based limiting procedure for unstructured cell-centered finite volumes with application to shallow water dynamics. *International Journal for Numerical Methods in Fluids*. 2013;71(5):584–633.
18. Delis A.I., Katsaounis T. Numerical solution of the two-dimensional shallow water equations by the application of relaxation methods. *Applied Mathematical Modelling*. 2005;29(8):754–783.
19. Ginting B.M., Mundani R.-P. Comparison of shallow water solvers: Applications for dam-break and tsunami cases with reordering strategy for efficient vectorization on modern hardware. *Water*. 2019;11(4):639.
20. Soares-Fraza S., Zech Y. Experimental study of dam-break flow against an isolated obstacle. *Journal of Hydraulic Research*. 2007;45(1):27–36.
21. Britov A. et al. Numerical Simulation of Propeller Hydrodynamics Using the Open Source Software. *International Conference on Computational Science*. Springer. 2023. P. 279–291.

Received 20.05.2024

Received 03.06.2024

Accepted 04.06.2024

About the Authors:

Ivan I. But, research engineer, Open-source Software Laboratory for Digital Modelling of Technical Systems, Ivannikov Institute for System Programming of the RAS (25, Aleksandr Solzhenitsyn st., Moscow, 109004, RF), Junior Researcher, Keldysh Institute of Applied Mathematics of RAS (4, Miusskaya sq., Moscow, 125047, RF), [ORCID](#), ivan.but@ispras.ru

Maria A. Kiryushina, research associate of Keldysh Institute of Applied Mathematics of Russian Academy of Sciences (KIAM RAS), Ivannikov Institute for System Programming of the Russian Academy of Sciences (ISP RAS) (25, Aleksandr Solzhenitsyn st., Moscow, 109004, RF), phd, [ORCID](#), [ResearcherID](#), m_ist@mail.ru

Stepan A. Elistratov, junior researcher, Shirshov Institute of oceanology of RAS (36, Nakhimovsky ave., Moscow, 117218, RF), research intern, Ivannikov Institute for system programming of RAS (25, Aleksandra Solzhenitsyna st., Moscow, 109004, RF), research engineer, Sobolev Institute of mathematics of Siberian branch of RAS (4, Akademika Koptuyuga ave., Novosibirsk, 630090, RF), [ORCID](#), sa.elist-ratov@yandex.ru

Tatiana G. Elizarova, chief researcher of Keldysh Institute of Applied Mathematics of Russian Academy of Sciences (KIAM RAS) (4, Miusskaya sq., Moscow, 125047, RF), doctor of physico-mathematical sciences, professor, [ORCID](#), [ResearcherID](#), [ScopusID](#), telizar@mail.ru

Artem D. Tiniakov, 5th year student of the Department of Information Transmission and Data Analysis Problems, Moscow Institute of Physics and Technology (9, Institutskiy pereulok, Dolgoprudny, 141701, RF), [ORCID](#), tiniakov.ad@gmail.com

Contributions of the co-authors:

All authors have made an equivalent contribution to the preparation of the publication.

Conflict of interest statement

The authors do not have any conflict of interest.

All authors have read and approved the final manuscript.

Поступила в редакцию 20.05.2024

Поступила после рецензирования 03.06.2024

Принята к публикации 04.06.2024

Об авторах:

Бут Иван Игоревич, инженер-исследователь лаборатории цифрового моделирования технических систем, Институт системного программирования им. В.П. Иванникова РАН (РФ, 109004, Москва, ул. Александра Солженицына, 25); младший научный сотрудник, Институт прикладной математики им. М.В. Келдыша РАН (РФ, 125047, Москва, Миусская пл., 4), [ORCID](#), ivan.but@ispras.ru

Кирюшина Мария Александровна, научный сотрудник ИПМ им. М.В. Келдыша РАН (РФ, 125047, Москва, Миусская пл., 4); ИСП им. В.П. Иванникова РАН (РФ, 109004, Москва, ул. Александра Солженицына, 25), кандидат физико-математических наук, [ORCID](#), [ResearcherID](#), m_ist@mail.ru

Елистратов Степан Алексеевич, младший научный сотрудник, лаборатория геофизической гидродинамики, Институт океанологии имени П.П. Ширшова РАН (РФ, 117218, Москва, Нахимовский пр., 65), стажер-исследователь, лаборатория цифрового моделирования технических систем, Институт системного программирования им. В.П. Иванникова РАН (РФ, 109004, Москва, ул. Александра Солженицына, 25); инженер-исследователь, отделение научно-исследовательской и проектной деятельности, Институт математики им. С.Л. Соболева СО РАН (РФ, 630090, Новосибирск, проспект Академика Коптюга, 4), [ORCID](#), sa.elist-ratov@yandex.ru

Елизарова Татьяна Геннадьевна, главный научный сотрудник ИПМ им. М.В. Келдыша РАН (125047, Москва, Миусская пл., 4), доктор физико-математических наук, профессор, [ORCID](#), [ResearcherID](#), [ScopusID](#), telizar@mail.ru

Тиняков Артём Дмитриевич, студент 5 курса кафедры проблем передачи информации и анализа данных, Московский физико-технический институт (национальный исследовательский университет) (РФ, 141701, Долгопрудный, Институтский переулок, 9), [ORCID](#), tiniakov.ad@gmail.com

Заявленный вклад соавторов:

Все авторы сделали эквивалентный вклад в подготовку публикации.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Все авторы прочитали и одобрили окончательный вариант рукописи.

MATHEMATICAL MODELLING МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ



Original Empirical Research



UDC 519.6

<https://doi.org/10.23947/2587-8999-2024-8-2-24-32>

Probabilistic Analysis of Heat Flux Distribution in the North Atlantic for 1979–2022

Konstantin P. Belyaev¹ , Andrey A. Kuleshov² ,
Anastasiya V. Novikova³, Natalia P. Tuchkova⁴

¹Shirshov Institute of Oceanology of Russian Academy of Sciences, Moscow, Russian Federation

²Keldysh Institute of Applied Mathematics of Russian Academy of Sciences, Moscow, Russian Federation

³Lomonosov Moscow State University Branch in Sevastopol, Sevastopol, Russian Federation

⁴Federal Research Center “Computer Sciences and Control” of Russian Academy of Sciences, Moscow, Russian Federation

andrew_kuleshov@mail.ru

Abstract

Introduction. The study of heat interaction processes and the distribution of heat fluxes in the oceans is important for understanding climate change on Earth. The North Atlantic, which is one of the key components of the global climate system, plays a significant role in regulating the climate of our latitudes. One of the key tools for analyzing heat distribution in the oceans is probabilistic analysis. In this work, using mathematical modelling methods, a statistical analysis of observational data on heat fluxes in the North Atlantic is carried out.

Materials and Methods. The used methods include the analysis of random processes specified by the stochastic differential equation (SDE) or the Ito equation, approximation of observational data, and solution of the Fokker-Planck-Kolmogorov (FPK) equation to describe the evolution of the probabilistic distribution of heat in the ocean.

Results. Using mathematical modelling methods, a probabilistic analysis of the distribution of heat fluxes in the North Atlantic for the period from 1979 to 2022 has been carried out. The results of the study made it possible to establish patterns of distribution of heat flux in the studied region over the period of time under consideration.

Discussion and Conclusions. The results may be useful for further study of climate processes in the North Atlantic, as well as for the development of resource management and environmental protection strategies.

Keywords: Random process, stochastic differential equation, Fokker-Planck-Kolmogorov equation, heat flow

Funding information. The research was carried out within the framework of a state assignment, topic No. FMWE-2024-0016.

For citation. Belyaev K.P., Kuleshov A.A., Novikova A.V., Tuchkova N.P. Probabilistic analysis of heat flux distribution in the North Atlantic for 1979–2022. *Computational Mathematics and Information Technologies*. 2024;8(2):24–32. <https://doi.org/10.23947/2587-8999-2024-8-2-24-32>

Оригинальное эмпирическое исследование

Вероятностный анализ распределения потоков тепла в Северной Атлантике за 1979–2022 годы

К.П. Беляев¹ , А.А. Кулешов² , А.В. Новикова³, Н.П. Тучкова⁴

¹Институт океанологии им. П.П. Ширшова Российской академии наук, г. Москва, Российская Федерация

²Институт прикладной математики им. М.В. Келдыша Российской академии наук, г. Москва, Российская Федерация

³Филиал Московского государственного университета имени М.В. Ломоносова в Севастополе, г. Севастополь, Российская Федерация

⁴Федеральный исследовательский центр «Информатика и управление» Российской академии наук, г. Москва, Российская Федерация

andrew_kuleshov@mail.ru

Аннотация

Введение. Изучение процессов теплообмена и распределения потоков тепла в океанах имеет важное значение для понимания климатических изменений на Земле. Северная Атлантика, являющаяся одним из ключевых компо-

нентов глобальной климатической системы, играет существенную роль в регулировании климата наших широт. Одним из ключевых инструментов для анализа распределения тепла в океанах является вероятностный анализ. В настоящей работе методами математического моделирования проводится статистический анализ данных наблюдений тепловых потоков в Северной Атлантике.

Материалы и методы. Используемые методы включают в себя анализ случайных процессов, заданных стохастическим дифференциальным уравнением (СДУ) или уравнением Ито, аппроксимацию данных наблюдений и решение уравнения Фоккера-Планка-Колмогорова (ФПК) для описания эволюции вероятностного распределения тепла в океане.

Результаты исследования. С помощью методов математического моделирования проведен вероятностный анализ распределения потоков тепла в Северной Атлантике за период с 1979 по 2022 годы. Результаты исследования позволили установить закономерности распределения потоков тепла в изучаемом регионе за рассматриваемый период времени.

Обсуждение и заключения. Полученные результаты могут быть полезными для дальнейшего изучения климатических процессов в Северной Атлантике, а также для разработки стратегий управления ресурсами и защиты окружающей среды.

Ключевые слова: случайный процесс, стохастическое дифференциальное уравнение, уравнение Фоккера-Планка-Колмогорова, поток тепла

Финансирование. Исследование выполнено в рамках государственного задания, тема № FMWE-2024-0016.

Для цитирования. Беляев К.П., Кулешов А.А., Новикова А.В., Тучкова Н.П. Вероятностный анализ распределения потоков тепла в Северной Атлантике за 1979–2022 годы. *Computational Mathematics and Information Technologies*. 2024;8(2):24–32. <https://doi.org/10.23947/2587-8999-2024-8-2-24-32>

Introduction. The study of heat exchange processes and the distribution of heat fluxes in the oceans is crucial for understanding climate changes on Earth. The North Atlantic, being one of the key components of the global climate system, plays a significant role in regulating the climate of our latitudes. In recent decades, there has been an increased interest in studying the temperature regime changes in this region and the interaction between the ocean and the atmosphere due to the phenomenon of global warming and its impact on climate processes in different parts of the planet.

One of the key tools for studying heat distribution in the oceans is probabilistic analysis. It allows for a quantitative assessment of the probability of various heat exchange scenarios and reveals trends in climate changes. In the North Atlantic, the heat flux plays a critical role in shaping the climate conditions of the region, thus conducting a probabilistic analysis of heat flux distribution for the period from 1979 to 2022 will provide a deeper understanding of the dynamics of climate processes in this key region.

There are numerous works in this direction; however, each new correct and meaningful study is of great interest from the standpoint of the methods used and the results of their application to the analysis of specific geophysical phenomena. Important recent studies in this area include [1]. The dynamics and spatial distribution of heat fluxes in the North Atlantic are well described, for instance, in [2, 3]. A statistical and quite detailed mathematical analysis of heat fluxes was previously carried out in [4].

In this study, a statistical analysis of observation data from the ERA5 archive [5] was conducted, and numerical solutions to the Fokker-Planck-Kolmogorov (FPK) equations were constructed based on statistically determined coefficients of the stochastic differential equation (SDE) (Ito equation) for interannual variability in the North Atlantic from 1979 to 2022. These results are new and original.

A data sample was taken from the ERA5 archive on the distribution of heat fluxes in the North Atlantic for the period from 1979 to 2022. Drift and diffusion coefficients for interannual variability of heat fluxes were then calculated, approximated by trigonometric polynomials, and finally, the Fokker-Planck-Kolmogorov equation was numerically solved with the approximated drift and diffusion coefficients under given initial conditions for the heat flux distribution function. The article concludes with an analysis of the results and their geophysical interpretation.

Materials and Methods

Mathematical Model. The variation in fluxes was modeled using a stochastic differential equation (Ito equation) [1, 6]:

$$dX = a(t, X)dt + b(t, X)dW, \quad (1)$$

where $X(t)$ is the heat flux at time t ; $dX = X(t + dt) - X(t)$ is the variability (increment) of the heat flux over time dt at a fixed point; $a(t, X)$, $b(t, X)$ are the drift and diffusion coefficients, respectively, which depend on time t and the values of the process $X(t)$; dW is the standard notation for Gaussian “white noise” with zero mean and unit variance, independent of the process $X(t)$. The data on heat fluxes specified at the nodes of a one-degree grid with a temporal resolution of 6 hours

from January 1, 1998, to December 31, 2022, were used in the study. The explicit and latent heat fluxes were calculated using the formulas:

$$Q_h = c_p \rho C_T (T_w - T_a) \mathbf{V}, \quad (2)$$

$$Q_e = L \rho C_e (Q_s - Q) \mathbf{V}. \quad (3)$$

In formulas (2) and (3), the heat fluxes are explicitly calculated through the following parameters at the ocean-atmosphere interface: water and air temperature T_w, T_a , wind speed modulus \mathbf{V} , specific humidity of the surface air Q and the maximum specific humidity at the ocean surface temperature Q_s . The coefficients of proportionality in relationships (2) and (3) are the heat exchange coefficient C_T (Schmidt number) and the moisture exchange coefficient C_e (Dalton number). Also present are the latent heat of evaporation L , the specific heat capacity of air at constant pressure c_p and its density ρ . In this study, the values of the fluxes Q_h, Q_e , were considered already known, and no additional calculations were required.

The study involves modelling the processes of heat flux variability in the North Atlantic and calculating the drift and diffusion coefficients at each moment in time using formulas (3). In this study, only heat flux maps for the middle of each month are provided.

Let the conditional probability be:

$$P(y|x) = P(X(t+dt) = y | X(t) = x),$$

if y, x are discrete, and the conditional probability density (Radon-Nikodym derivative) is $p(y|x) = p(X(t+dt) = y | x < X(t) = x + dx)$ and if the process $X(t)$ is considered continuous. For definiteness, we will consider the process continuous with the conditional probability density $p(y|x)$. To distinguish the process $X(t)$ from its values, the latter will be denoted by lowercase letters. To determine the coefficients $a(t, x), b(t, x)$ the following formulas are used [6]:

$$a(t, x) = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{+\infty} (y - x) p(y|x) dy, \quad (4)$$

$$b^2(t, x) = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_{-\infty}^{+\infty} (y - x)^2 p(y|x) dy. \quad (5)$$

Thus, to determine the coefficients $a(t, x), b(t, x)$ according to formulas (4) and (5), it is necessary to perform the algorithm described in [1]. The range of values $X(t)$ from $X_{\min}(t) = X_{\min}$ до $X_{\max}(t) = X_{\max}$ to $X_i = X_{\min} + i * \frac{X_{\max} - X_{\min}}{L}$, $i = 0, \dots, L$ into L subintervals $[X_i, X_{i+1}]$, with $X_i < X(t) = x < X_{i+1}$ and all grid points falling into this interval are fixed. Let there be m (such points (the intervals should be chosen so that $m > 0$ for any $X_i < X(t) = x < X_{i+1}$)). Moving to the step $t + dt$ a similar division of the range of values $X(t)$ into subintervals $[Y_i, Y_{i+1}]$ is made, and from the previously fixed points, only those points are selected for which $Y_i < X(t + dt) = y < Y_{i+1}$ for any y . Let there be l such points. Then $P(y|x) = l/m$. Next, according to formulas (4) and (5), the coefficients $a(t, x)$ and $b^2(t, x)$ are calculated.

Fokker-Planck-Kolmogorov Parabolic Equation. As indicated in the previous paragraph, the variability of the random process is represented as (1). Expression (1) is understood in the integral sense, i. e.

$$X(t + \Delta t) - X(t) = \int_t^{t+\Delta t} a(u, X) du + \int_t^{t+\Delta t} b(u, X) [W(u + du) - W(u)].$$

The coefficients $a(t, x)$ and $b^2(t, x)$ are calculated according to formulas (4) and (5). Consequently, the equation for calculating the probabilities of the flux values at a given (climatic) moment in time can be written as

$$\frac{\partial p}{\partial t} = -\frac{\partial(a(t, x)p)}{\partial x} + \frac{1}{2} \frac{\partial^2(b^2(t, x)p)}{\partial x^2}, \quad (6)$$

where $p(t, x)$ is the sought probability density at moment t for the flux value x , and the other notations are as above. Equation (6) (Fokker-Planck-Kolmogorov equation) is solved under Sommerfeld boundary conditions and a given initial probability distribution. Analytically, this problem is generally not solvable, but numerically its solution is not particularly difficult. However, for strongly oscillating coefficients, the numerical solution of this equation leads to significant computational errors. Therefore, to reduce computational errors, it makes sense to smooth the strongly oscillating coefficients $a(t, x)$ and $b^2(t, x)$.

Least Squares Method for Approximating Monthly Coefficients. The coefficient $a(t, x)$ is approximated as a trigonometric function:

$$\hat{a}(t, x) = A(x) \sin(wt) + B(x) \cos(wt) + C(x). \quad (7)$$

This form of the approximating function was chosen because the original drift coefficients and diffusion coefficients exhibit a pronounced intra-annual oscillatory variability $w = 2\pi/12$ (m^{-1}). Thus, this form of approximation is physically justified.

The parameters $A(x)$, $B(x)$, $C(x)$ were determined sequentially using the least squares method. First, the parameter $C(x)$ was determined as the mean value of the sample:

$$\frac{1}{518} \sum_{t=1}^{518} a(t, x) - C(x) = 0, \quad (8)$$

and then the parameters $A(x)$ and $B(x)$ were determined. This algorithm is well-known in practical applications and does not require additional justification. After all amplitudes are determined using formula (7), the approximation of the coefficient $a(t, x)$ is represented as:

$$\hat{a}(t, x) = A(x) \sin(wt + \varphi(x)) + C(x). \quad (9)$$

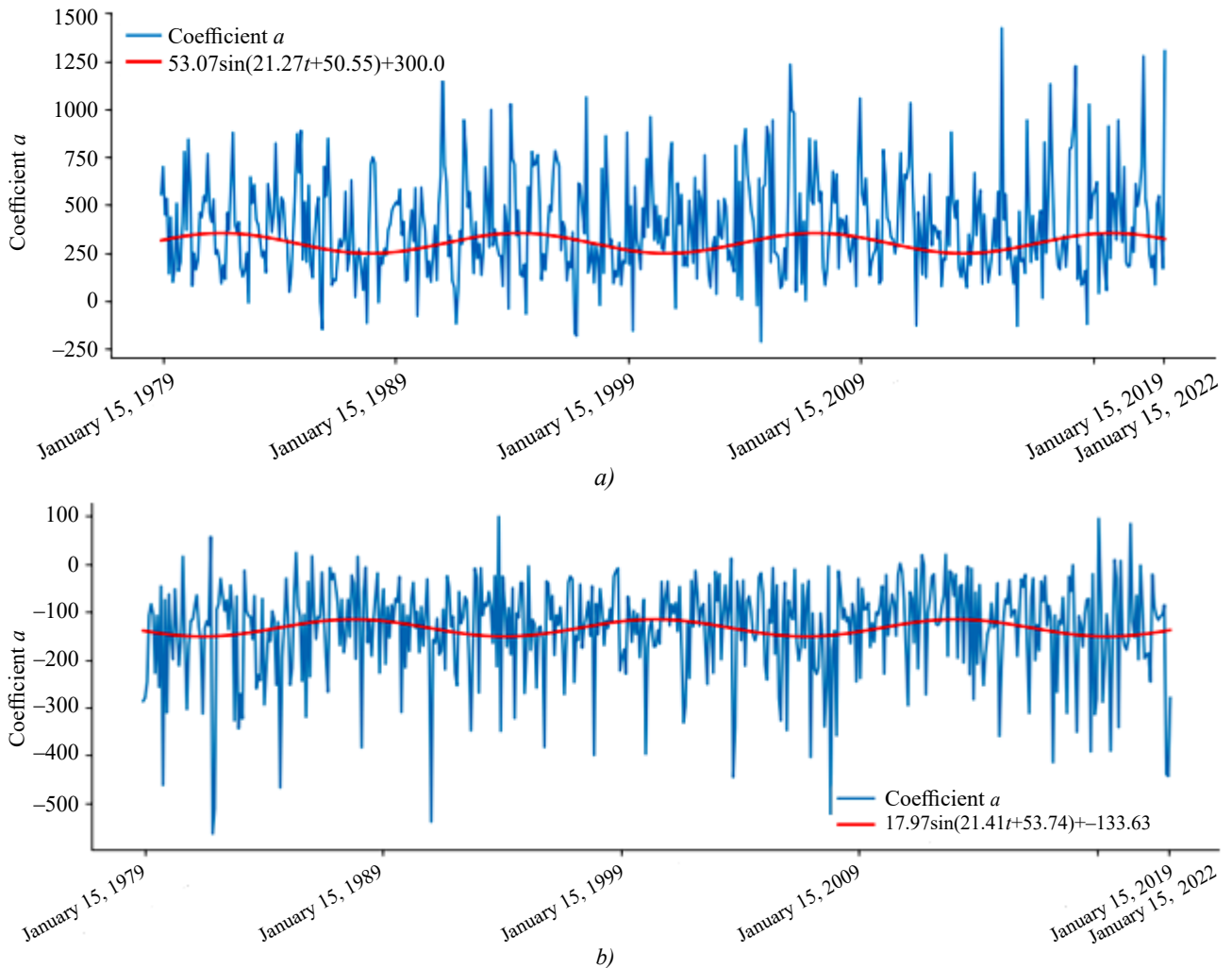
The same procedure was applied to approximate the diffusion coefficient $b(t, X)$

$$\hat{b}(t, x) = D(x) \sin(wt + \psi(x)) + F(x). \quad (10)$$

Results

Analysis of FPK Equation Coefficients. For equation (6), an implicit difference scheme of second-order approximation [7] with the smoothed coefficients given by formulas (9) and (10) was implemented.

Figure 1 shows the graphs of the coefficients $a(t, X)$ and $\hat{a}(t, x)$ averaged over a month for a period of 40 years. The smoothed curves $\hat{a}(t, x)$, constructed according to formulas (7), with the corresponding amplitudes and phases found by the least squares method, are shown in red. Figure 1 illustrates the changes in the coefficients $a(t, X)$ and $\hat{a}(t, x)$ over time at flux values X , equal to X_{\max} (Fig. 1 a), X_{\min} (Fig. 1 b) and the average value for the region (Fig. 1 c).



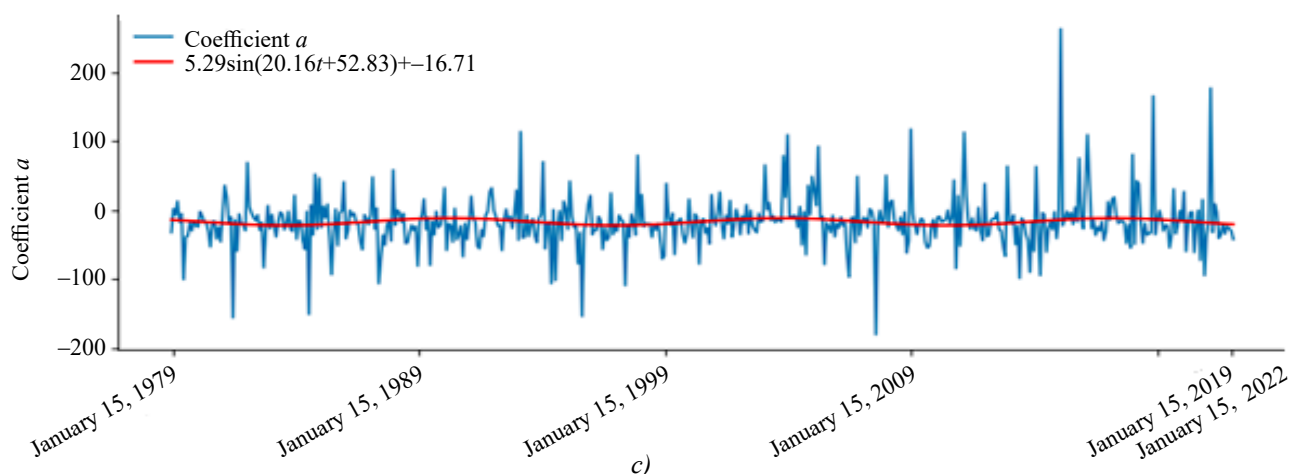
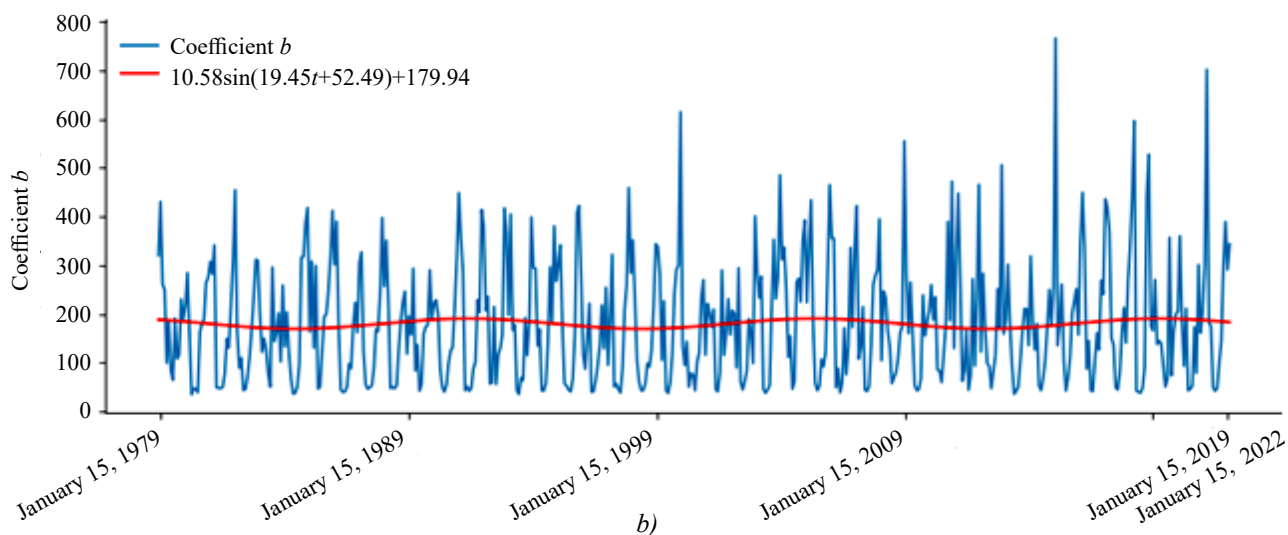
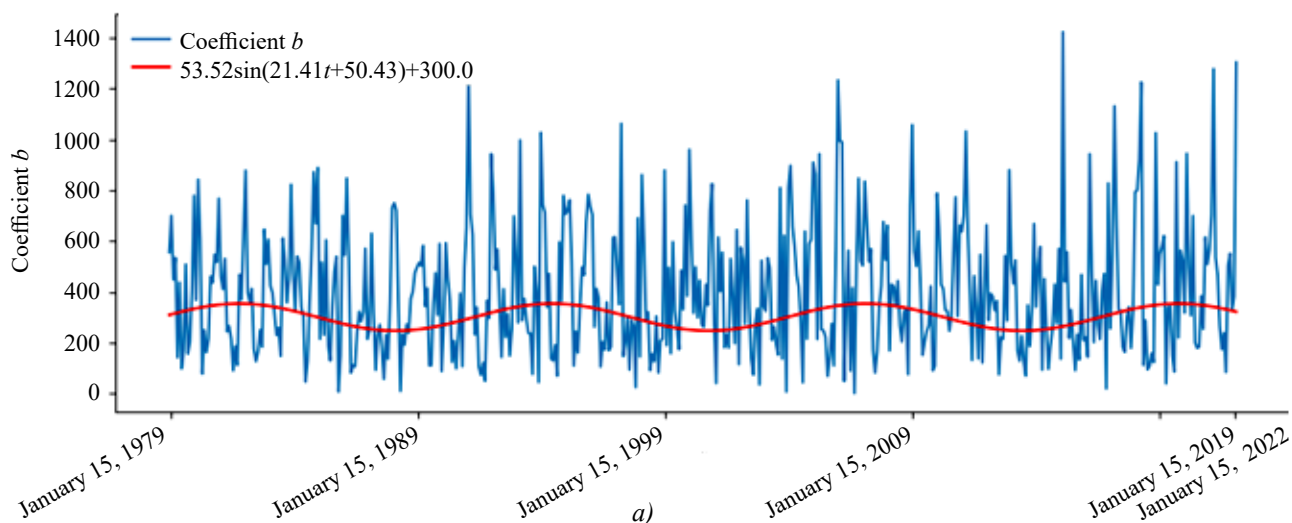


Fig. 1. Changes over time of the drift coefficient $a(t, X)$ (W/m²/day) and the smoothed coefficient $\hat{a}(t, x)$ (red curves) over 40 years: a — at flux values X_{\max} , b — at flux values X_{\min} , c — at average flux values X

The graphs have a reasonable physical interpretation. In the first graph, the predominance of upward “peaks” and the attainment of the maximum value of the drift coefficient are noticeable. In the second graph, the “peaks” point downwards, and the drift coefficient reaches a minimum. In the third graph, a scatter of values from 200 to –200 is noticeable, with the coefficients themselves averaging around zero.

Similarly, the graphs of the diffusion coefficient $b(t, X)$ and $\hat{b}(t, x)$ look similar (Fig. 2).



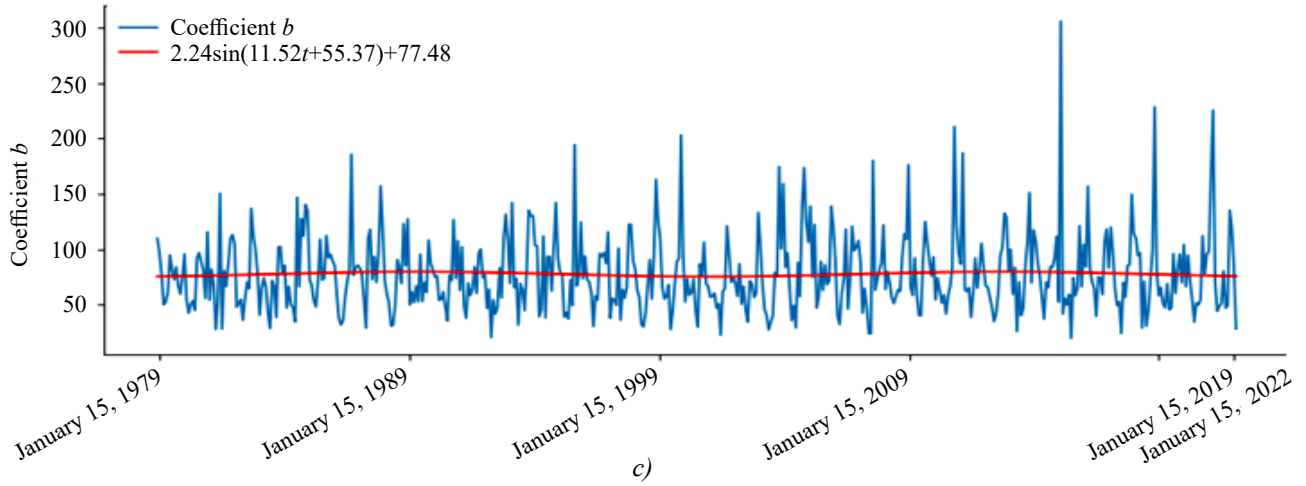
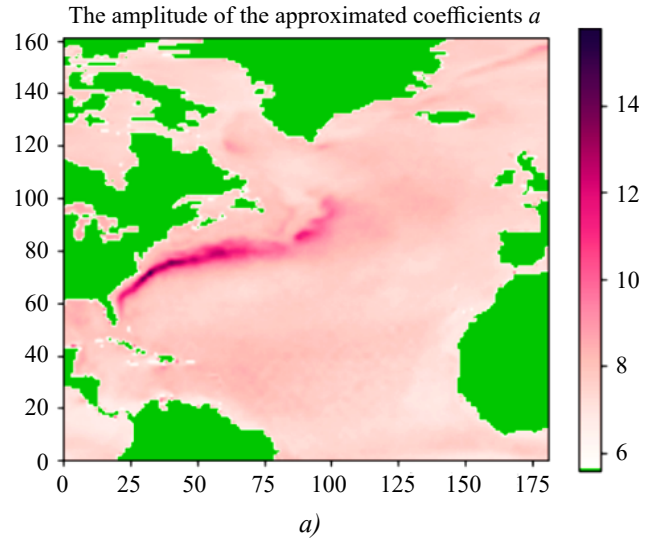


Fig. 2. Changes over time of the diffusion coefficient $b(t, x)$ (W/m²/day) and the smoothed coefficient $\hat{b}(t, x)$ (red curves) over 40 years: a — at flux values X_{\max} , b — at flux values X_{\min} , c — at average flux values X

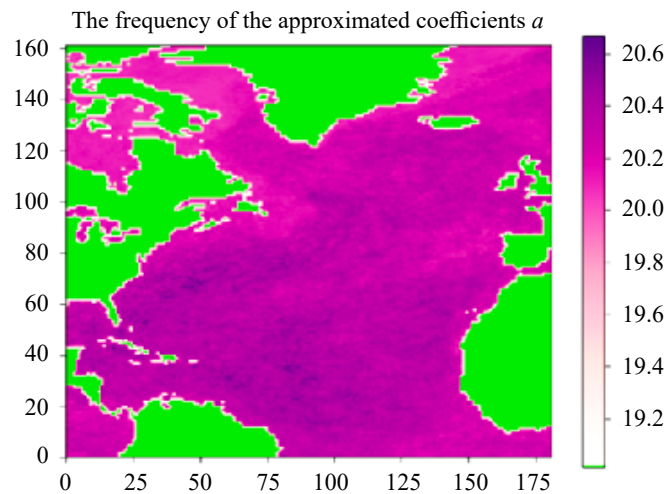
Based on the calculated values of amplitude, phase, and shift in formula (9) (as functions of only spatial variables), their maps were constructed. A ready-made map mask of the North Atlantic was used for this purpose.

The maps clearly show the Gulf Stream and the North Atlantic Current (northeast of Iceland), where local maxima of amplitude (Fig. 3 a), phase (Fig. 3 b), and shift (Fig. 3 c) are noticeable.

The maps for the amplitude, phase, and shift of the diffusion coefficient $\hat{b}(t, x)$ (see formula (10)) have a similar appearance, and thus, we will not include them here.



$a)$



$b)$

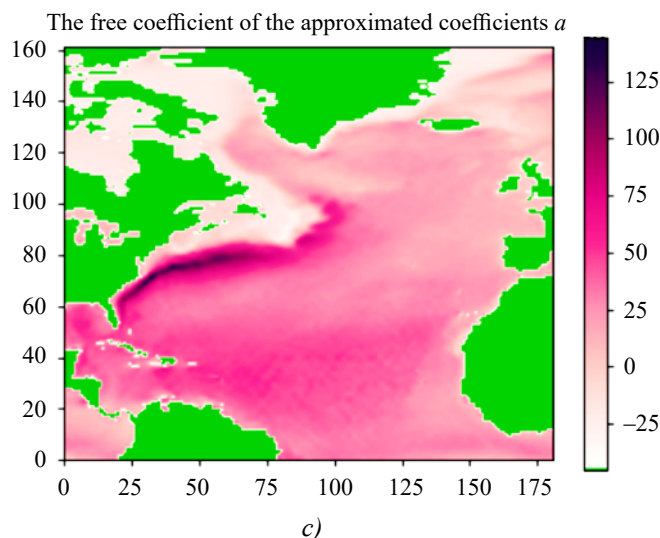


Fig. 3. Spatial distribution for $\hat{a}(t, x)$: a — amplitudes, b — phases, c — shifts

Results of Numerical Calculations of the FPK Equation and Their Analysis. Figure 4 shows the results of the numerical solution of the FPK equation on January 15 of the years 1979, 1989, 1999, 2009, 2019, and 2022 for the average flux value on the corresponding date. The initial condition was set according to the formula $p(0, x) = 1 / \left[(2\pi)^{1/2} \sigma \right] \exp(-x^2 / 2\sigma^2)$ (Gaussian initial density), where σ was set as the difference between the maximum and minimum flux values.

From these figures, it is evident that the density plots are generally reasonable, accurately reflecting the dynamics of the equation's coefficients. Their maximum value oscillates from 0.18 to 0.01, and their average value also oscillates from 0 (at the beginning and end of the calculation) to 50 (months) in the middle of the calculations.

The graphs are not symmetrical, and the distribution of the FPK equation solution significantly differs from the normal distribution. The pulsations of the curves at their upper parts are explained by the coarse spatial resolution of the grid (one degree-approximately 100 km) on which the equation's coefficients are located. Nevertheless, these distributions are physically reasonable and can be used in applications for analyzing climate data and making climate predictions.

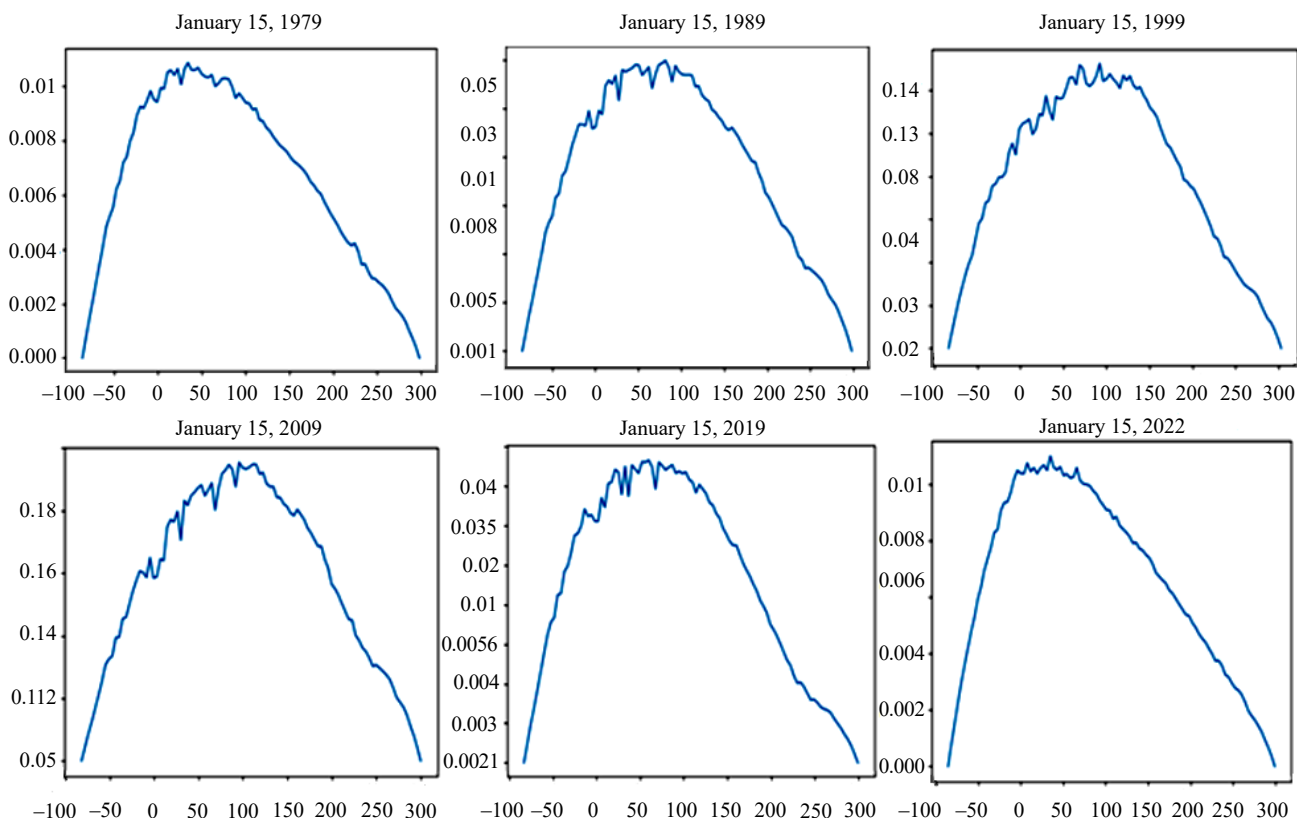


Fig. 4. Solution of the Fokker-Planck-Kolmogorov equation with approximated coefficients

Discussion and Conclusion. In this study, probabilistic analysis of heat flux distribution in the North Atlantic from 1979 to 2022 was conducted using mathematical modelling methods. The research results allowed for the establishment of quantitative and qualitative patterns of heat flux distribution in the studied region over the considered time period. The obtained results can be useful for further studying climate processes in the North Atlantic and for developing resource management and environmental protection strategies.

It should be noted that this research has its limitations, including the limited availability of data and the assumptions underlying the applied models. Further research in this area may include extending the time range and improving analysis methods to obtain more accurate predictions and interpretations, such as investigating multidimensional (interdependent) quantities and finding their joint probability distributions.

References

1. Gorshenin A., Osipova A., Belyaev K. Stochastic analysis of air-sea heat fluxes variability in the North Atlantic in 1979–2022 based on reanalysis data. *Computers & Geosciences*. 2023;181(10):105461. <https://doi.org/10.1016/j.cageo.2023.105461>
2. Cayan D.R. Variability of latent and sensible heat fluxes estimated using bulk formulate. *Atmosphere-Ocean*. 1992;30(1):1–42. <https://doi.org/10.1080/07055900.1992.9649429>
3. Parfitt R., Czaja A., Kwon Y.-O. The impact of SST resolution change in the ERA Interim reanalysis on wintertime Gulf Stream frontal air-sea interaction. *Geophysical Research Letters*. 2017;44(7):3246–3254. <https://doi.org/10.1007/s00376-020-0072-0>
4. Belyaev K., Gorshenin A., Korolev V., Osipova A. Comparison of Statistical Approaches for Reconstructing Random Coefficients in the Problem of Stochastic Modelling of Air-Sea Heat Flux Increments. *Mathematics*. 2024;12(2):228. <https://doi.org/10.3390/math12022288>
5. Simmons A., Hersbach H., Munoz-Sabater J., Nicolas J., Vamborg F., Berrisford P., de Rosnay P., Willett K., Woollen J. Low frequency variability and trends in surface air temperature and humidity from ERA5 and other datasets. *ECMWF Technical Memoranda*. 2021;881. <https://doi.org/10.21957/ly5vbtbfd>
8. Gikhman I.I., Skorokhod A.V. *Introduction to the Theory of Random Processes*. Dover Publications, 1996. 516 p. (In Russ.).
9. Samarsky A.A. *The Theory of Difference Schemes*. CRC Press, 2001. 786 p. (In Russ.). <https://doi.org/10.1201/9780203908518>

Received 04.04.2024

Received 17.04.2024

Accepted 22.04.2024

About the Authors:

Konstantin P. Belyaev, Leading Researcher at the P.P. Shirshov Institute of Oceanology of the Russian Academy of Sciences (36, Nakhimovsky Prospekt, Moscow, 117997, RF), [MathNet](#), [ScopusID](#), [ORCID](#), kosbel55@gmail.com

Andrey A. Kuleshov, Chief Researcher at the M.V. Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences (4, Miusskaya Sq., Moscow, 125047, RF), [MathNet](#), [ScopusID](#), [ORCID](#), andrew_kuleshov@mail.ru

Anastasia V. Novikova, Student at the Sevastopol Branch of Moscow State University named after M.V. Lomonosov (7, Heroes of Sevastopol St., Sevastopol, 299001, RF), novikovaav1@my.msu.ru

Natalia P. Tuchkova, Senior Researcher at the Federal Research Center “Informatics and Management” of the Russian Academy of Sciences (40, Vavilova St., Moscow, 119333, RF), [MathNet](#), [ScopusID](#), [ORCID](#), natalia_tuchkova@mail.ru

Contributions of the co-authors:

P.B. Konstantin — problem statement, development of research methods, analysis of results.

A.A. Kuleshov — problem statement, analysis of results.

A.V. Novikova — numerical calculations, visualization of results.

N.P. Tuchkova — preparation of geophysical data.

Conflict of interest statement

The authors do not have any conflict of interest.

All authors have read and approved the final manuscript.

Поступила в редакцию 04.04.2024

Поступила после рецензирования 17.04.2024

Принята к публикации 22.04.2024

Об авторах:

Беляев Константин Павлович, ведущий научный сотрудник Института океанологии им. П.П. Ширшова Российской академии наук (РФ, 117997, Москва, Нахимовский просп., 36), [MathNet](#), [ScopusID](#), [ORCID](#), kosbel55@gmail.com

Кулешов Андрей Александрович, главный научный сотрудник Института прикладной математики им. М.В. Келдыша Российской академии наук (РФ, 125047, Москва, Миусская пл., 4), [MathNet](#), [ScopusID](#), [ORCID](#), andrew_kuleshov@mail.ru

Новикова Анастасия Вячеславовна, студент Филиала Московского государственного университета имени М.В. Ломоносова в Севастополе (РФ, 299001, Севастополь, ул. Героев Севастополя, 7), novikovaav1@my.msu.ru

Тучкова Наталия Павловна, старший научный сотрудник Федерального исследовательский центр «Информатика и управление» Российской академии наук (РФ, 119333, Москва, ул. Вавилова, 40), [MathNet](#), [ScopusID](#), [ORCID](#), natalia_tuchkova@mail.ru

Заявленный вклад соавторов:

К.П. Беляев — постановка задачи, разработка методов исследования, анализ результатов.

А.А. Кулешов — постановка задачи, анализ результатов.

А.В. Новикова — проведение численных расчетов, визуализация результатов.

Н.П. Тучкова — подготовка геофизических данных.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Все авторы прочитали и одобрили окончательный вариант рукописи.

MATHEMATICAL MODELLING МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ



Original Empirical Research



UDC 519.6

<https://doi.org/10.23947/2587-8999-2024-8-2-33-44>

Mathematical Modelling of Catastrophic Surge and Seiche Events in the Azov Sea Using Remote Sensing Data

Elena A. Protsenko¹ , Natalya D. Panasenko^{1,2} , Sofia V. Protsenko¹

¹Taganrog Institute named after A.P. Chekhov (branch) of RSUE, Taganrog, Russian Federation

²Don State Technical University, Rostov-on-Don, Russian Federation

✉ capros@rambler.ru

Abstract

Introduction. This work is devoted to the mathematical modelling of extreme sea level fluctuations in the Azov Sea using remote sensing data. The aim of the study is to develop and apply a mathematical model that allows more accurate prediction of surge and seiche events caused by extreme wind conditions. The relevance of the work is due to the need to improve the forecasts of hydrodynamic processes in shallow water bodies (such as the Azov Sea), where such phenomena can have significant economic and ecological consequences. The goal of this work is to develop and apply a mathematical model for predicting extreme sea level fluctuations in the Azov Sea caused by wind conditions.

Materials and Methods. The study is based on the analysis of remote sensing data and observations of wind speed and direction over the Azov Sea. The primary method used is mathematical modelling, which includes solving the system of shallow water hydrodynamics equations. Wind condition data were collected from November 20 to 25, 2019, during which catastrophic sea level fluctuations were observed. The model considers the components of water flow velocity, water density, hydrodynamic pressure, gravitational acceleration, and turbulence exchange coefficients.

Results. The modelling showed that prolonged easterly winds with speeds up to 22 m/s led to significant surge and seiche fluctuations in sea level. The maximum amplitudes of fluctuations were recorded in the central part of the Taganrog Bay, where the wind direction and speed remained almost constant throughout the observation period. Data from various platforms located in different parts of the Azov Sea confirmed a significant decrease in water level in the northeast and an increase in the southwest.




Discussion and Conclusions. The study results confirm that using mathematical models in combination with remote sensing data allows more accurate predictions of extreme sea level fluctuations. This is important for developing measures to prevent and mitigate the consequences of surge and seiche events in coastal areas. In the future, it is necessary to improve models by including additional factors such as climate change and anthropogenic impact on the Azov Sea ecosystem.

Keywords: mathematical modelling, sea level fluctuations, remote sensing data, surge and seiche events, hydrodynamics

Funding information. The study was supported by the Russian Science Foundation grant no. 23-21-00210, <https://rscf.ru/project/23-21-00210/>

For citation. Protsenko E.A., Panasenko N.D., Protsenko S.V. Mathematical modelling of catastrophic surge and seiche events in the Azov Sea using remote sensing data. *Computational Mathematics and Information Technologies*. 2024;8(2):33–44. <https://doi.org/10.23947/2587-8999-2024-8-2-33-44>

Математическое моделирование катастрофических сгонно-нагонных явлений Азовского моря с использованием данных дистанционного зондирования

Е.А. Проценко¹ , Н.Д. Панасенко^{1,2} , С.В. Проценко¹ 

¹Таганрогский институт имени А.П. Чехова (филиал) РГЭУ (РИНХ), г. Таганрог, Российская Федерация

²Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация

✉ capros@rambler.ru

Аннотация

Введение. Работа посвящена математическому моделированию экстремальных колебаний уровня Азовского моря с использованием данных дистанционного зондирования. Цель исследования заключается в разработке и применении математической модели, которая позволяет более точно прогнозировать сгонно-нагонные явления, вызванные экстремальными ветровыми условиями. Актуальность работы обусловлена необходимостью улучшения прогнозов гидродинамических процессов в мелководных водоемах (таких, как Азовское море), где подобные явления могут иметь значительные экономические и экологические последствия. Цель данной работы — разработка и применение математической модели для прогнозирования экстремальных колебаний уровня Азовского моря, вызванных ветровыми условиями.

Материалы и методы. Исследование основывается на анализе данных дистанционного зондирования и наблюдений за скоростью и направлением ветра над Азовским морем. В качестве основного метода используется математическое моделирование, включающее решение системы уравнений волновой гидродинамики для мелководного водоема. Данные о ветровых условиях были собраны в период с 20 по 25 ноября 2019 года, когда наблюдались катастрофические колебания уровня моря. Модель учитывает компоненты скорости водного потока, плотность водной среды, гидродинамическое давление, ускорение свободного падения и коэффициенты турбулентного обмена.

Результаты исследования. Моделирование показало, что продолжительное действие восточного ветра со скоростью до 22 м/с привело к значительным сгонно-нагонным колебаниям уровня моря. Максимальные амплитуды колебаний были зафиксированы в центральной части Таганрогского залива, где направление и скорость ветра оставались практически неизменными в течение всего периода наблюдений. Данные с различных платформ, расположенных в разных частях Азовского моря, подтвердили наличие значительного снижения уровня воды на северо-востоке и повышения на юго-западе.

Обсуждение и заключения. Результаты исследования подтверждают, что использование математических моделей в сочетании с данными дистанционного зондирования позволяет более точно прогнозировать экстремальные колебания уровня моря. Это имеет важное значение для разработки мер по предупреждению и минимизации последствий сгонно-нагонных явлений в прибрежных районах. В дальнейшем необходимо совершенствовать модели, включая дополнительные факторы, такие как изменение климатических условий и антропогенное воздействие на экосистему Азовского моря.

Ключевые слова: математическое моделирование, колебания уровня моря, данные дистанционного зондирования, сгонно-нагонные явления, гидродинамика

Финансирование. Исследование выполнено за счет гранта Российского научного фонда № 23-21-00210, <https://rscf.ru/project/23-21-00210/>

Для цитирования. Проценко Е.А., Панасенко Н.Д., Проценко С.В. Математическое моделирование катастрофических сгонно-нагонных явлений Азовского моря с использованием данных дистанционного зондирования. *Computational Mathematics and Information Technologies*. 2024;8(2):33–44. <https://doi.org/10.23947/2587-8999-2024-8-2-33-44>

Introduction. The Azov Sea, being a shallow water body, is subject to significant water level fluctuations under the influence of wind conditions. These fluctuations can have serious economic and ecological consequences for coastal areas, including flooding and infrastructure damage. Despite the importance of understanding and predicting such phenomena, existing models do not always account for all necessary factors, leading to insufficient accuracy in forecasts [1–5].

The aim of this study is to develop and apply a mathematical model to predict extreme fluctuations in the Azov Sea level caused by wind conditions. The relevance of the study is driven by the need to create accurate and reliable tools for predicting surge and seiche events, which will enable the development of effective measures to prevent and mitigate their negative consequences. Given the increasing frequency of extreme weather events due to climate change, the task of developing such models becomes particularly important.

Catastrophic surge and seiche events in the Azov Sea pose a serious threat to coastal settlements and the region's ecology. Predicting and preventing these events requires mathematical modelling. For the effective operation of these

models, not only field data but also remote sensing data (RSD) of the Earth are needed. In this work, the authors consider various approaches to the mathematical modelling of surge and seiche events in the Azov Sea based on remote sensing data.

The scientific novelty of the work lies in the use of remote sensing data combined with advanced mathematical modelling methods. This allows for a more accurate accounting of dynamic changes in the water environment and wind conditions, enhancing the predictive capability of the models. The work presents new approaches to approximating turbulence exchange coefficients and organizing computational experiments, which significantly improves the accuracy and reliability of the obtained results.

Materials and Methods

1. Research Object. Throughout the year, weak winds predominate over the Azov Sea. Their occurrence rate is 60–70 %, the share of moderate winds is 20 %, and strong winds about 10%. Winds with speeds of 20–24 m/s can occur at any time of the year, while those over 24 m/s are only observed from October to April. Winds over 14 m/s predominantly have a northeast and east direction. From November 20 to November 25, 2019, the Azov Sea experienced catastrophic surge and seiche fluctuations caused by the prolonged action of easterly winds. An analysis of the wind rose, calculated for the central part of the Taganrog Bay from November 19 to 25, shows that the wind direction and speed remained almost constant (Fig. 1). The maximum wind speed reached 19–22 m/s, while the prevailing winds for the coast and open part of the Azov Sea are 4.5–5.5 m/s on the coast and 7.5 m/s in the central part. Data from November 27, 2019, indicated a gradual decrease in wind speed and, consequently, a gradual decrease in the amplitude of surge and seiche fluctuations.

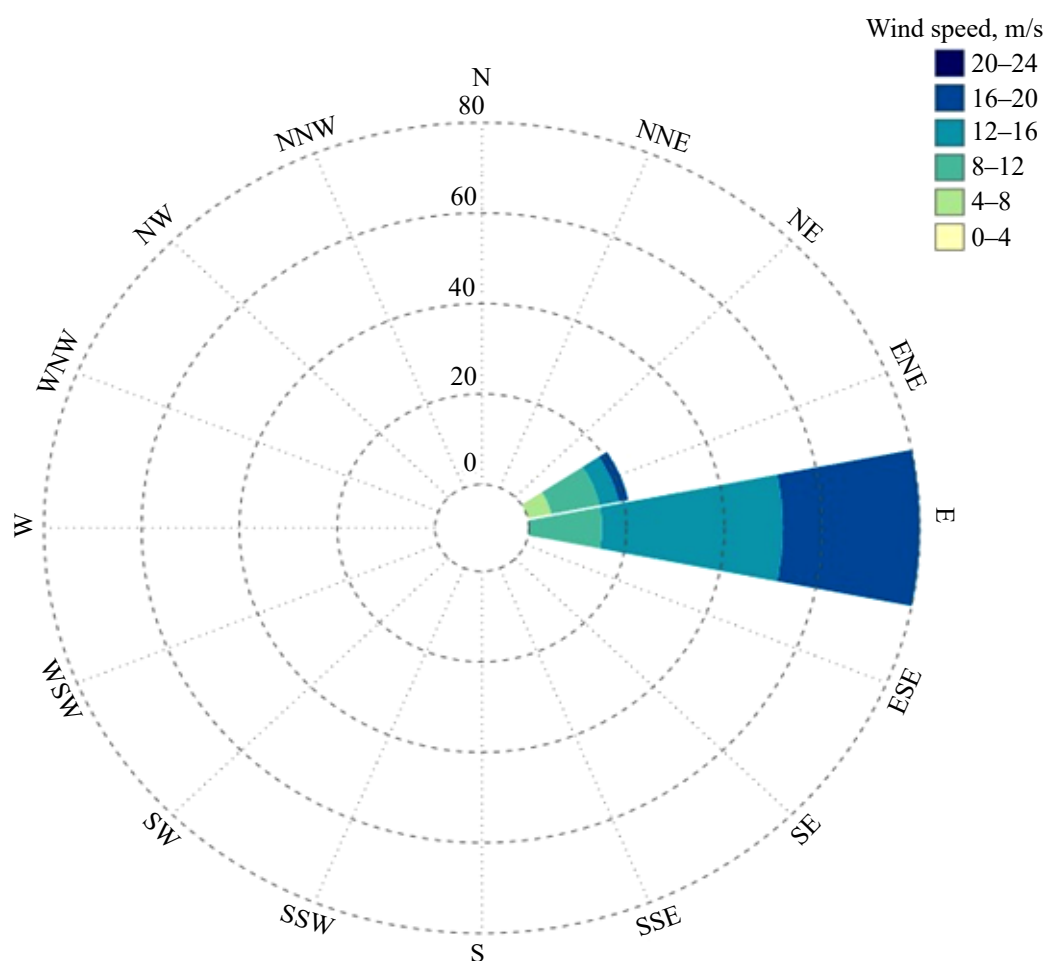


Fig. 1. Wind rose for the central part of the Taganrog Bay from November 19 to 25, 2019

Wind speed primarily depends on the barometric gradient and the characteristics of the underlying surface. The latter factor causes a significant increase in the average wind speed over the open sea compared to the average wind speed on the coast. This increase is due to the low friction of the air flow over the water surface, resulting in wind speeds recorded at coastal stations being somewhat lower than those recorded offshore.

The coastal array of monthly average sea level values includes monthly average data on sea level measured by a gauge at coastal hydrometeorological stations in the Azov Sea. Data from four platforms located in the northeastern part of the sea (Taganrog, Ochakov Spit, Yeisk Port, Dolzhanskaya) show a significant decrease in water level in November 2019 [6].

Data from the Mysovoye platform, located at Cape Kazantip in the southwestern part of the sea, show a diametrically opposite situation — a significant rise in water level was observed in November.



Fig. 2. Depth map of the Azov Sea with marked hydrometeorological stations: Taganrog (1), Ochakov Spit (2), Yeisk Port (3), Dolzhanskaya (4), Mysovoye (5)

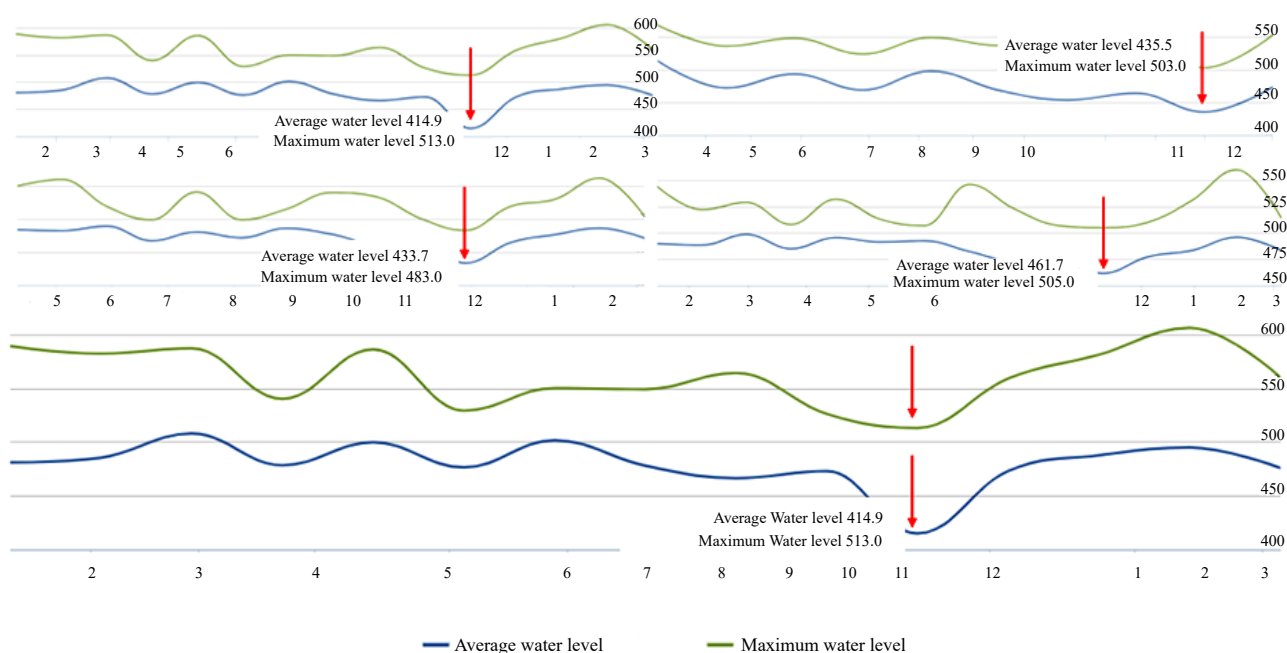


Fig. 3. Sea level at coastal hydrometeorological stations Taganrog, Ochakov Spit, Yeisk Port, Dolzhanskaya in 2019

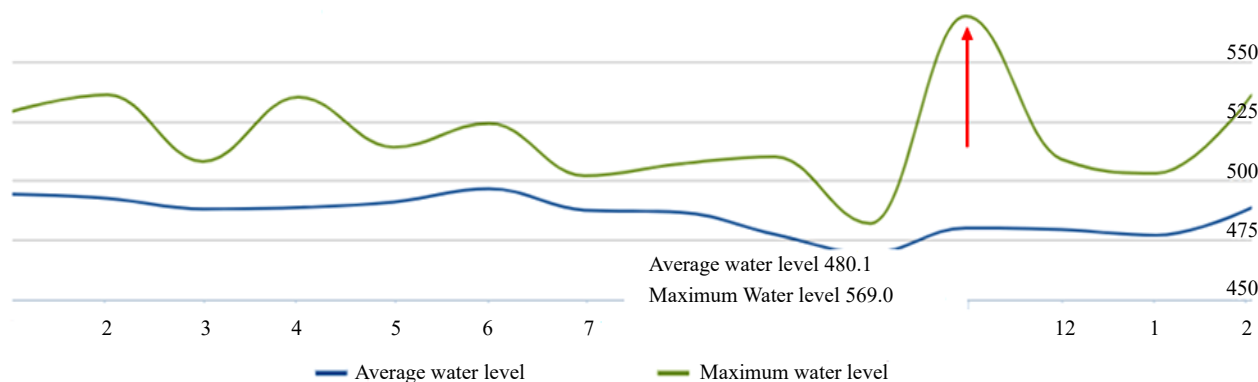


Fig. 4. Climate data on hydrometeorological conditions of the coastal zone of the Azov Sea: sea level at coastal hydrometeorological station Mysovoye in 2019

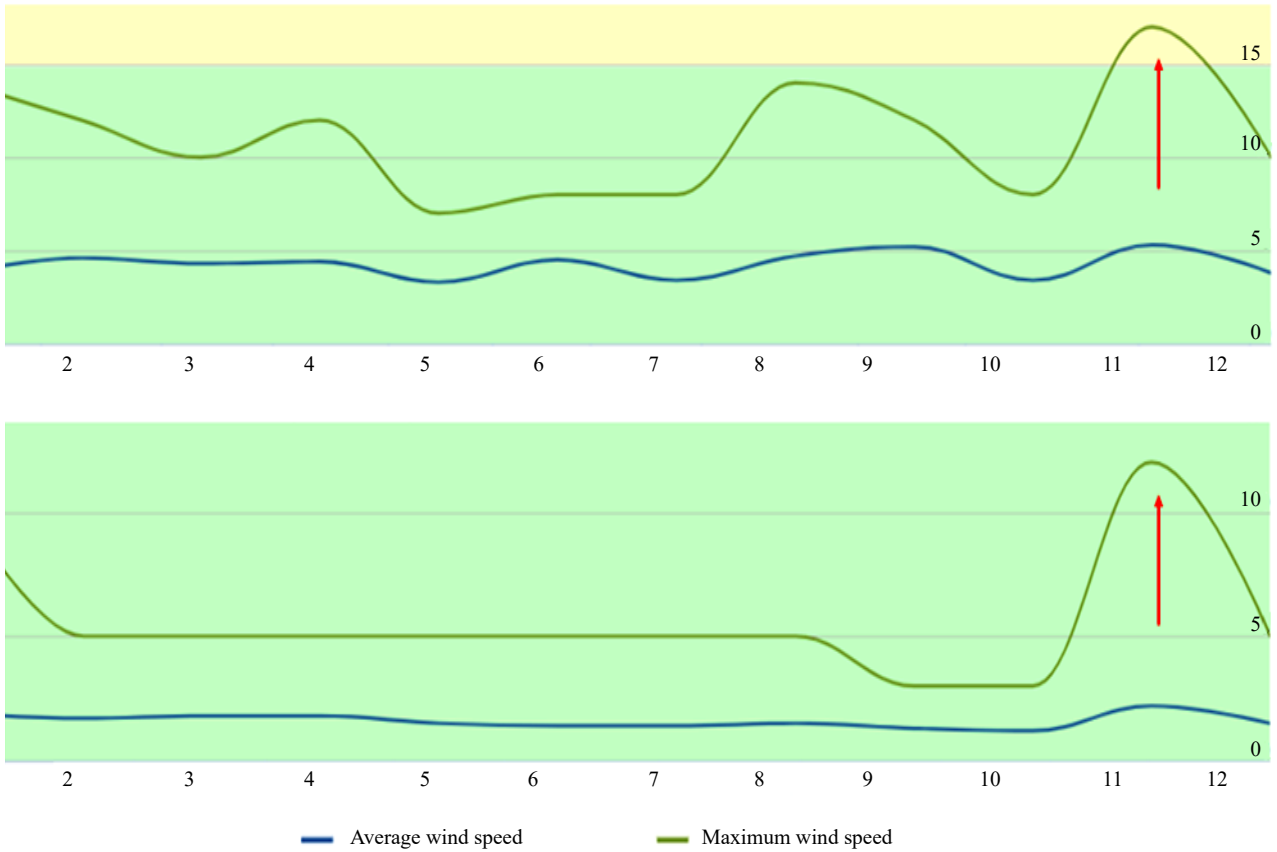


Fig. 5. Average and maximum wind speeds in 2019 in the northeastern part of the sea (top graph) and southwestern part (bottom graph) of the Azov Sea

2. Problem Statement. Modelling of surge and seiche processes is based on solving the system of shallow water wave hydrodynamics equations (written in Cartesian coordinates x, y, z) [7–8]:

$$\begin{aligned} u'_t + uu'_x + vv'_y + ww'_z &= -\frac{1}{\rho} P'_x + (\mu u'_x)'_x + (\mu v'_y)'_y + (\mu w'_z)'_z, \\ v'_t + uv'_x + vv'_y + ww'_z &= -\frac{1}{\rho} P'_y + (\mu v'_x)'_x + (\mu v'_y)'_y + (\mu w'_z)'_z, \end{aligned} \quad (1)$$

$$\begin{aligned} w'_t + uw'_x + vv'_y + ww'_z &= -\frac{1}{\rho} P'_z + (\mu w'_x)'_x + (\mu w'_y)'_y + (\mu w'_z)'_z + g, \\ \rho'_t + (\rho u)'_x + (\rho v)'_y + (\rho w)'_z &= 0. \end{aligned} \quad (2)$$

In equations (1) and (2), the following notations are used: $\xi'_x = \frac{\partial \xi}{\partial x}$ u, v, w are the components of the water flow velocity in a shallow water body; ρ is the water density; P is the hydrodynamic pressure; g is the gravitational acceleration; μ, ν are the coefficients of turbulent exchange in the horizontal and vertical directions, respectively.

Let $\tau = \rho_a C_d_s |\mathbf{w}| \mathbf{w}$ be the tangential stress vector for the free surface, where C_d_s is the dimensionless surface resistance coefficient, which depends on wind speed ($C_d_s = 0.0026$), \mathbf{w} is the wind speed vector relative to the water, and ρ_a is the atmospheric density. For the bottom, the tangential stress vector is given considering the water movement $\tau = \rho_a C_d_b |\mathbf{V}| \mathbf{V}$, $C_d_b = gk^2/h^{1/3}$, k is the group roughness coefficient ($k = 0.04$), $h = H + \eta$ is the depth of the water area, H is the depth to the undisturbed surface, and η is the height of the free surface relative to sea level.

We use an approximation that allows constructing a non-uniform depth-dependent vertical turbulent exchange coefficient based on measured flow velocity fluctuations:

$$\nu = C_s^2 \Delta^2 \frac{1}{2} \sqrt{\left(\frac{\partial \bar{u}}{\partial z} \right)^2 + \left(\frac{\partial \bar{v}}{\partial z} \right)^2}, \quad (3)$$

where C_s is a dimensionless empirical constant determined based on the calculation of the decay process of homogeneous isotropic turbulence, Δ is the characteristic grid scale, and \bar{u}, \bar{v} are the time-averaged fluctuations of the water flow velocity components in the horizontal direction.

To build a discrete mathematical model of the posed hydrodynamics problem and its numerical implementation, we introduce a uniform grid:

$$\bar{w}_h = \{t^n = n\tau, x_i = ih_x, y_j = jh_y, z_k = kh_z; n = \overline{0, N_t}, i = \overline{0, N_x}, j = \overline{0, N_y}, k = \overline{0, N_z}, \\ N_t\tau = T, N_x h_x = l_x, N_y h_y = l_y, N_z h_z = l_z\},$$

where τ is the time step; h_x, h_y, h_z are the spatial steps; N_t is the number of time layers; T is the upper boundary of the time coordinate; N_x, N_y, N_z are the number of nodes along the spatial coordinates; l_x, l_y, l_z are the lengths of the edges of the elementary parallelepiped in the directions of the axes O_x, O_y and O_z respectively.

To solve the hydrodynamics problem, we will use the pressure correction method. The method variant in the case of variable density takes the form [9–10]:

$$\begin{aligned} \frac{\tilde{u} - u}{\tau} + u\bar{u}'_x + v\bar{u}'_y + w\bar{u}'_z &= (\mu\bar{u}'_x)'_x + (\mu\bar{u}'_y)'_y + (\mu\bar{u}'_z)'_z, \\ \frac{\tilde{v} - v}{\tau} + u\bar{v}'_x + v\bar{v}'_y + w\bar{v}'_z &= (\mu\bar{v}'_x)'_x + (\mu\bar{v}'_y)'_y + (\mu\bar{v}'_z)'_z, \\ \frac{\tilde{w} - w}{\tau} + u\bar{w}'_x + v\bar{w}'_y + w\bar{w}'_z &= (\mu\bar{w}'_x)'_x + (\mu\bar{w}'_y)'_y + (\mu\bar{w}'_z)'_z + g, \\ p''_{xx} + p''_{yy} + p''_{zz} &= \frac{\hat{p} - p}{\tau^2} + \frac{(\hat{p}\tilde{u})'_x}{\tau} + \frac{(\hat{p}\tilde{v})'_y}{\tau} + \frac{(\hat{p}\tilde{w})'_z}{\tau}, \\ \frac{\hat{u} - \tilde{u}}{\tau} &= -\frac{1}{\rho}\hat{p}'_x, \quad \frac{\hat{v} - \tilde{v}}{\tau} = -\frac{1}{\rho}\hat{p}'_y, \quad \frac{\hat{w} - \tilde{w}}{\tau} = -\frac{1}{\rho}\hat{p}'_z. \end{aligned} \quad (4)$$

Here: u, v, w are the components of the velocity vector $V = \{u, v, w\}$; $\{\hat{u}, \hat{v}, \hat{w}\}, \{\tilde{u}, \tilde{v}, \tilde{w}\}$ are the components of the velocity fields at the new and intermediate time layers, respectively; $\bar{u} = (\tilde{u} + u)/2$; \hat{p} and p are the distributions of the water density at the new and previous time layers.

3. Combination of Local Binary Patterns (LBP) and Neural Networks for Initial and Boundary Conditions, and Model Verification. When correctly formulating initial-boundary value problems for systems of nonlinear partial differential equations and determining various functional dependencies in constructing mathematical models, it is essential to have real input data (boundary conditions, initial conditions, information about source functions). During decision-making regarding risks associated with hazardous natural phenomena and disasters, up to 50 % of the total time spent on computer modelling and forecasting can be dedicated to recognizing the specific situation. This is particularly relevant to determining the location and size of phenomena such as surge, seiche, and hypoxic events.

A readily available source of in situ information for mathematical modelling can be Earth remote sensing data. The recognition and input of this data as initial and boundary conditions is a complex and labor-intensive procedure that requires the creation of specialized algorithms [9–10]. Recognition is also indispensable for subsequent comparative analysis of the accuracy of mathematical modelling.

The “neural network-LBP” algorithm, described in detail in [11–13], is a domestic solution in this problem area. For further work, we will use satellite images taken on November 17 and 22, 2019 (Fig. 6 and 7).

The composition of various image pixels collectively works to determine the orientation and scale of the object in the image. This allows for the recognition and localization of necessary features regardless of the image’s rotation or changes in brightness/contrast. Image segmentation is performed considering a predefined set of semantic classes, which are encoded by numbers from 0 to a certain limit. These semantic labels consist of subsets, each corresponding to a specific semantic label. Thus, when a pixel is labeled with a certain semantic class number, its corresponding instance identifier does not matter. This means that all pixels in a class belong to one instance (e. g., the same color).

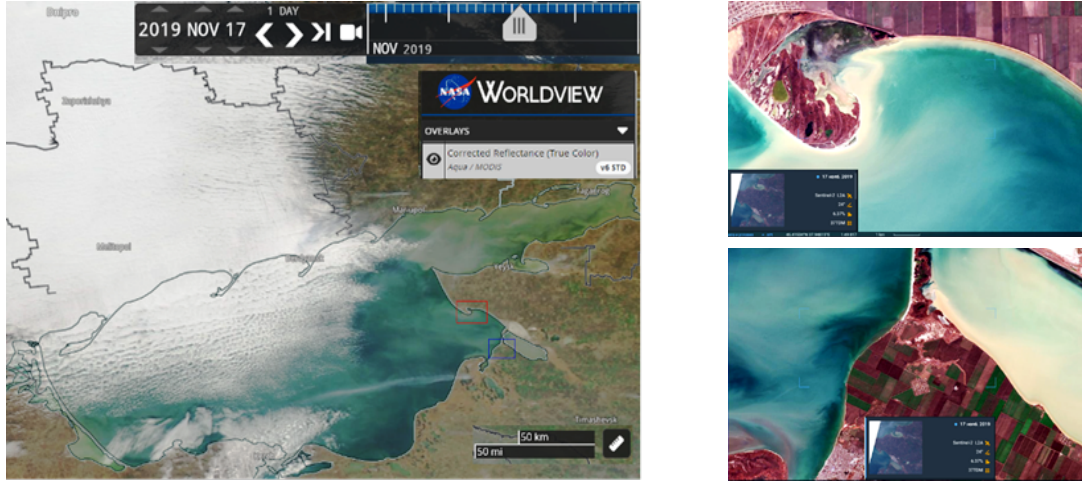


Fig. 6. Images of the study area from WorldView [14] and Sentinel-2 L2A [13] satellites, 17.11.2019

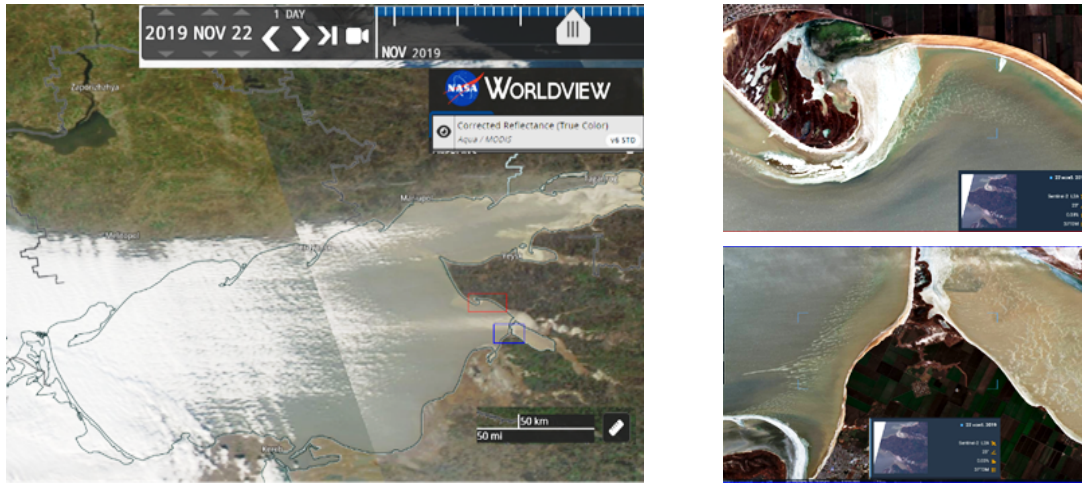


Fig. 7. Images of the study area from WorldView [14] and Sentinel-2 L2A [15] satellites, 22.11.2019

In the algorithm's operation, segments are matched, and the PQ is calculated considering matches in two stages. In the first stage, empirical data and predicted segments are matched with a threshold SP of 0.5. This approach results in unique matches where segments do not overlap, so the maximum number of matches per segment is one. In the second stage, the PQ is calculated considering the predicted and ground truth segmentation of the image. Each ground truth segment can have no more than one corresponding predicted segment with an SP strictly greater than 0.5, and vice versa. This means that if a corresponding predicted segment is found for a particular ground truth segment, it automatically falls into the “matched pairs” category. If no predicted segments correspond to a ground truth segment, it falls into the “unmatched ground truth segments” category. If no predicted segment is found for a particular ground truth segment, the predicted segments that intersect it are classified as “unmatched predicted segments”. Thus, after matching, each segment falls into one of three sets: matched pairs (MP), unmatched predicted segments (UP), and unmatched ground truth segments (UT).

$$PQ = \frac{\sum_{(\phi, gg) \in MP} SP(\phi, gg)}{|MP| + \frac{1}{2}|UP| + \frac{1}{2}|UT|}, \quad (5)$$

where ϕ are the predicted segments; gg are the ground truth segments (object environment); SP is the threshold value equal to 0.5; MP are the matched pairs; UP are the unmatched predicted segments; UT are the unmatched ground truth segments.

4. Description of the Software Package. In this work, an advanced software package was used, which takes into account the dynamic changes of the computational area due to wave processes, jet effects, and the multicomponent nature of impurities. This package is designed to construct three-dimensional velocity fields of water movement, considering

a depth-heterogeneous vertical turbulent exchange coefficient and dynamic restructuring of the computational area geometry. The software package is implemented in C++. The algorithm for organizing a computational experiment based on the developed software package is presented in Fig. 8.

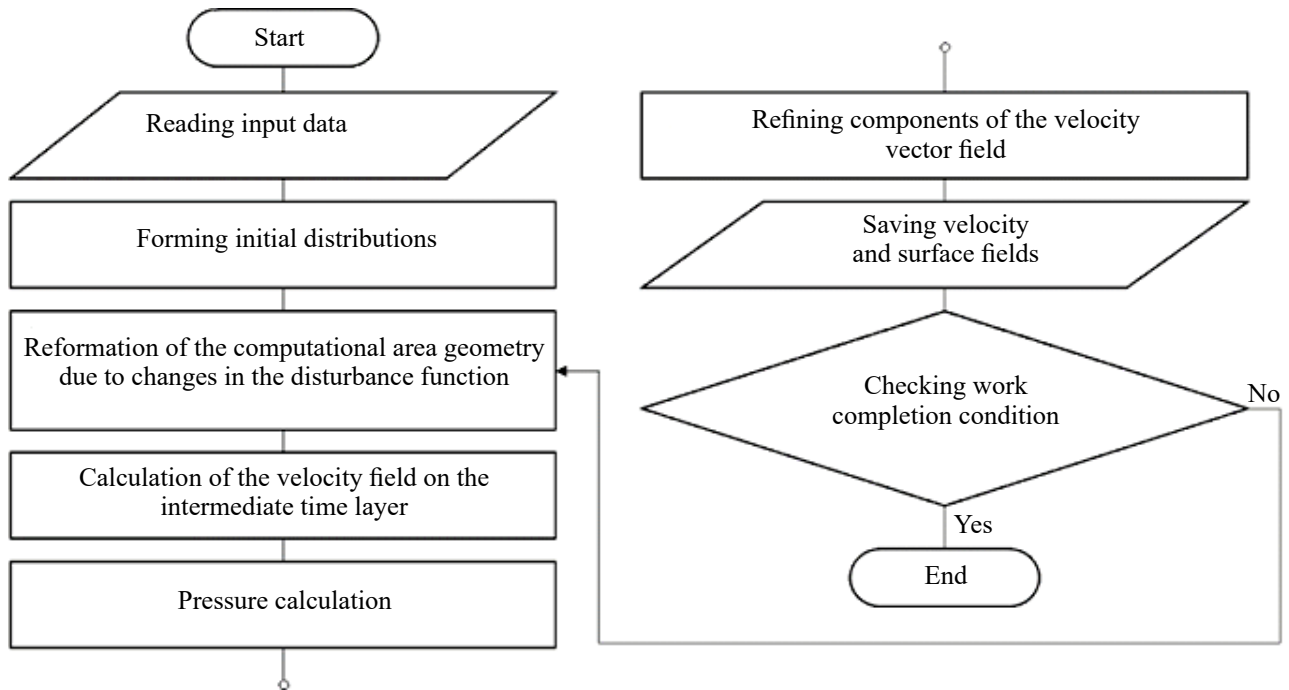


Fig. 8. Algorithm scheme of computational experiment organization

The software allows for the specification of complex bottom geometry in the form of a raster model, built based on known cadastral survey data and remote sensing data, as well as the type and characteristics of the oscillation source, wind direction, and speed.

Research results

1. Numerical experiments based on a three-dimensional model of wave hydrodynamics. Prolonged action of wind at significant speeds led to the development of wind-driven oscillations, where extreme downwelling was observed in the eastern part of the Azov Sea, and upwelling in the western part. During the storm event in the Taganrog Bay overnight from November 21 to 22, 2019, the sea level dropped below 220–240 cm relative to the mean level. Maximum upwelling sea level rises were observed in the coastal western part of the Azov Sea, reaching 130–140 cm.

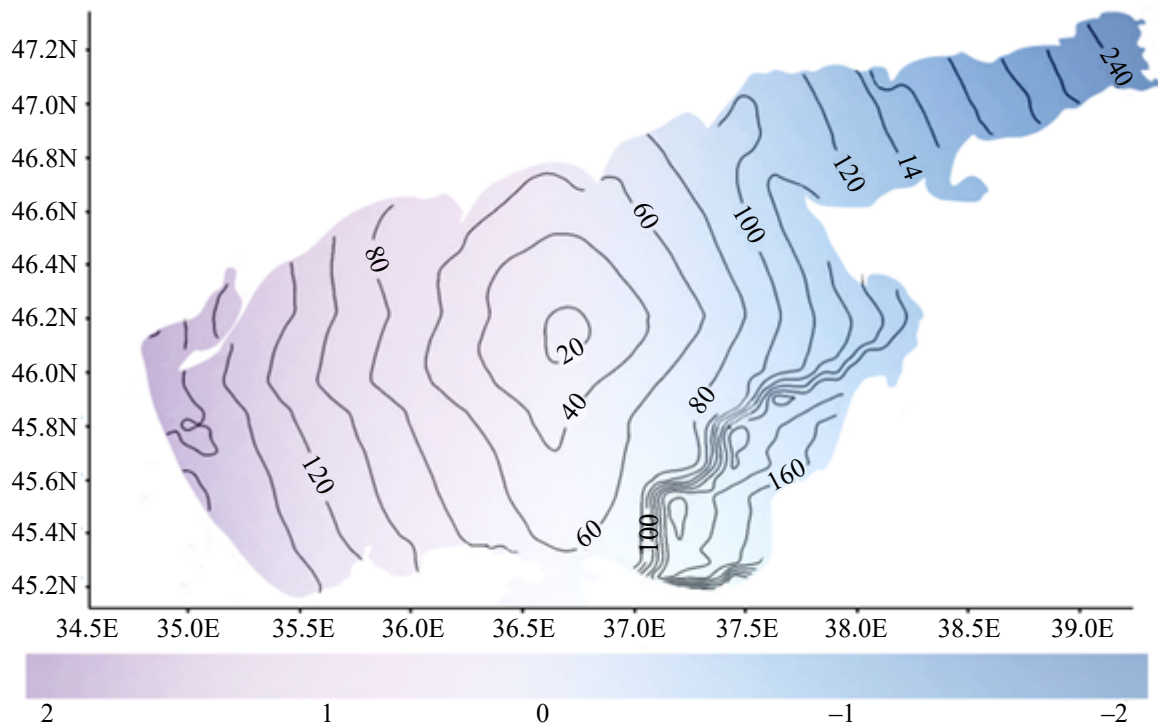


Fig. 8. Sea level relative to the mean level of the Azov Sea, November 22, 2019

For modelling, a section of Taganrog Bay measuring 100 by 50 meters was selected, with a maximum depth of 1.8 meters at this site. Calculations utilized a grid of $100 \times 200 \times 40$ computational nodes, with a time step of 0.01 seconds. Wind speed during modelling was set at 15 m/s from the east.

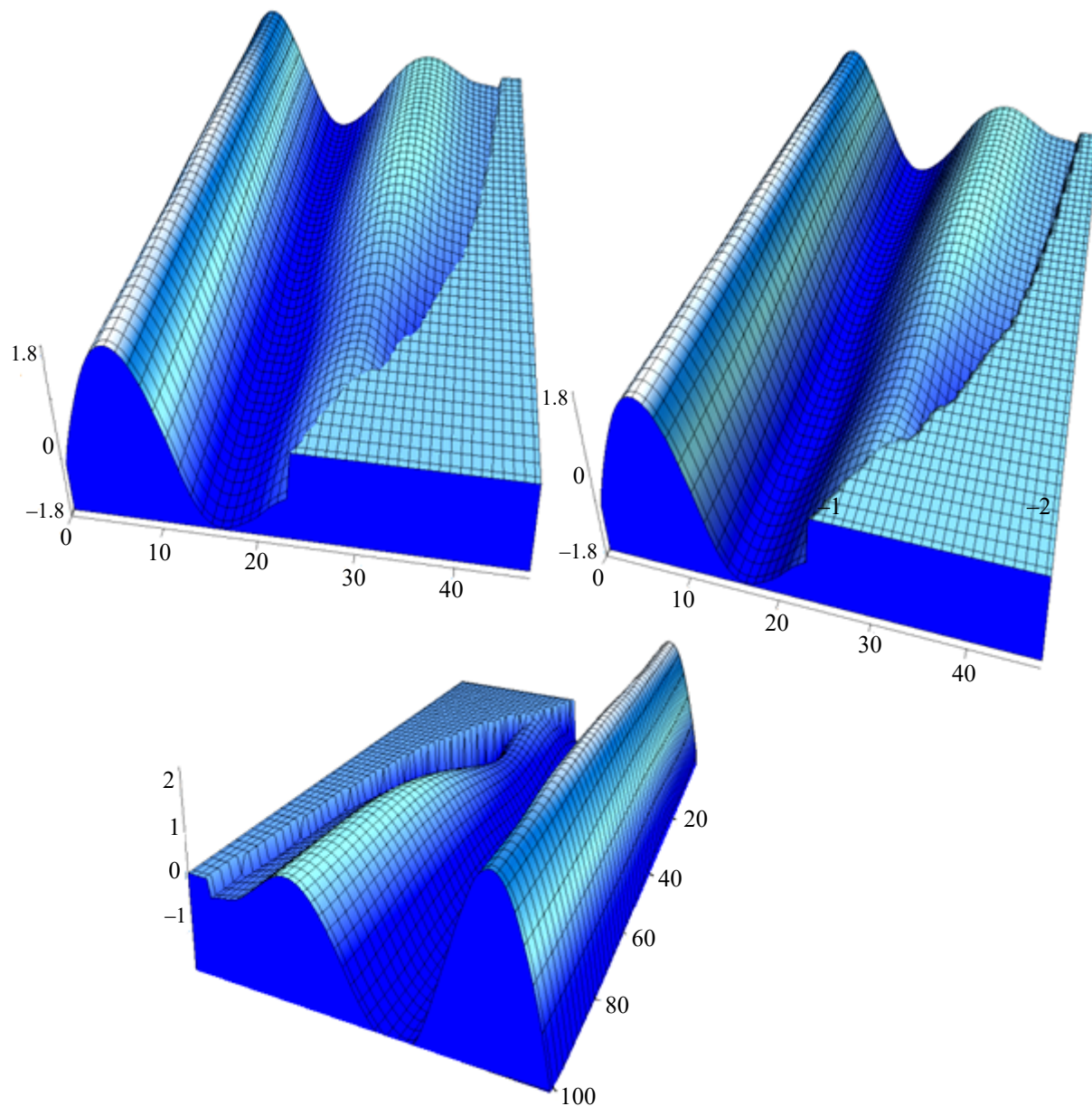


Fig. 10. Modelling results of wave heights with an east wind speed of 15 m/s

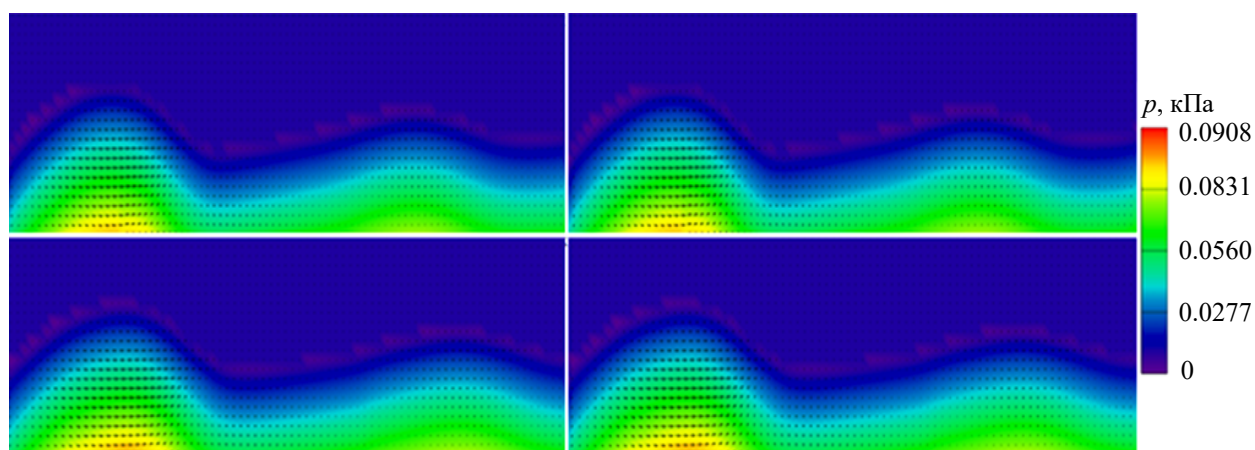


Fig. 10. Modelling results of velocity vector field and pressure with an easterly wind at a speed of 15 m/s

According to data from the Federal Service for Hydrometeorology and Environmental Monitoring, on November 22, 2019, the Sea of Azov experienced a northeast wind at speeds of 12–17 m/s, gusting up to 27 m/s, with wave heights of 1.3–1.8 m. During November 22 daytime, overnight, and morning of November 23, a northeast wind at speeds of 11–16 m/s, gusting up to 22 m/s, produced wave heights of 1–2 m. Modelling results are consistent with the data from the Federal Service for Hydrometeorology and Environmental Monitoring. During established wave conditions, modeled wave heights ranged from 1.6 to 2 meters.

2. Results of numerical experiments based on the “neural network-LBP” algorithm for processing boundary contours of aquatic environments using RSD. Results of the software module allow tracking the dynamics of contours (in our case, the shoreline) over an extended period of time (Fig. 11 and 12).

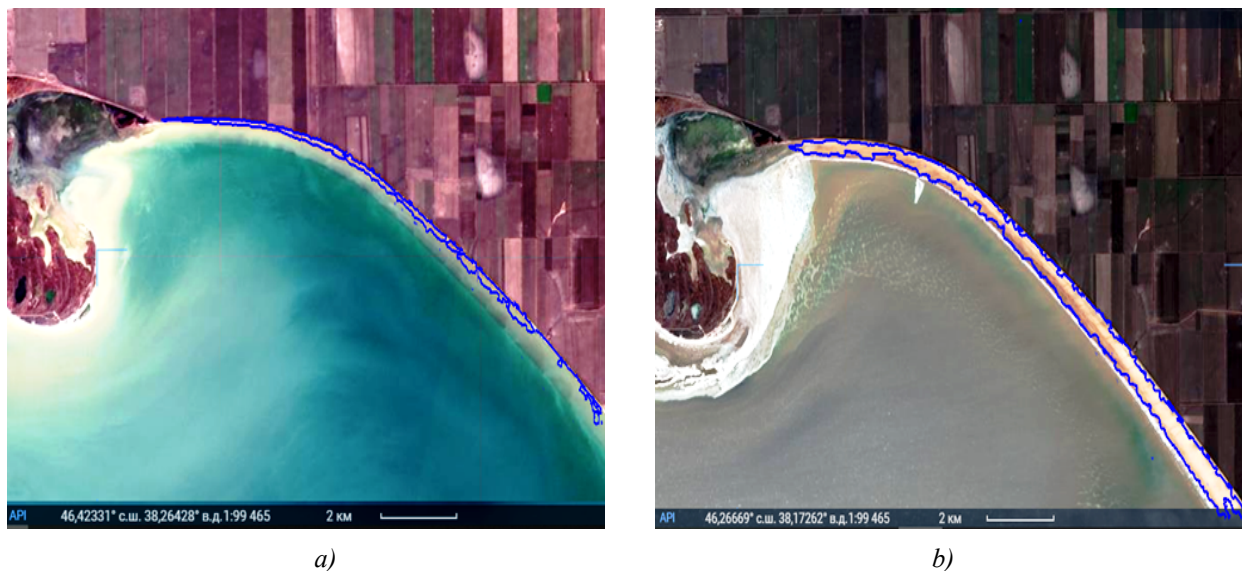


Fig. 11. Snapshots of the study area — Stanitsa Yasenskaya:
a — November 17, 2019; b — November 22, 2019

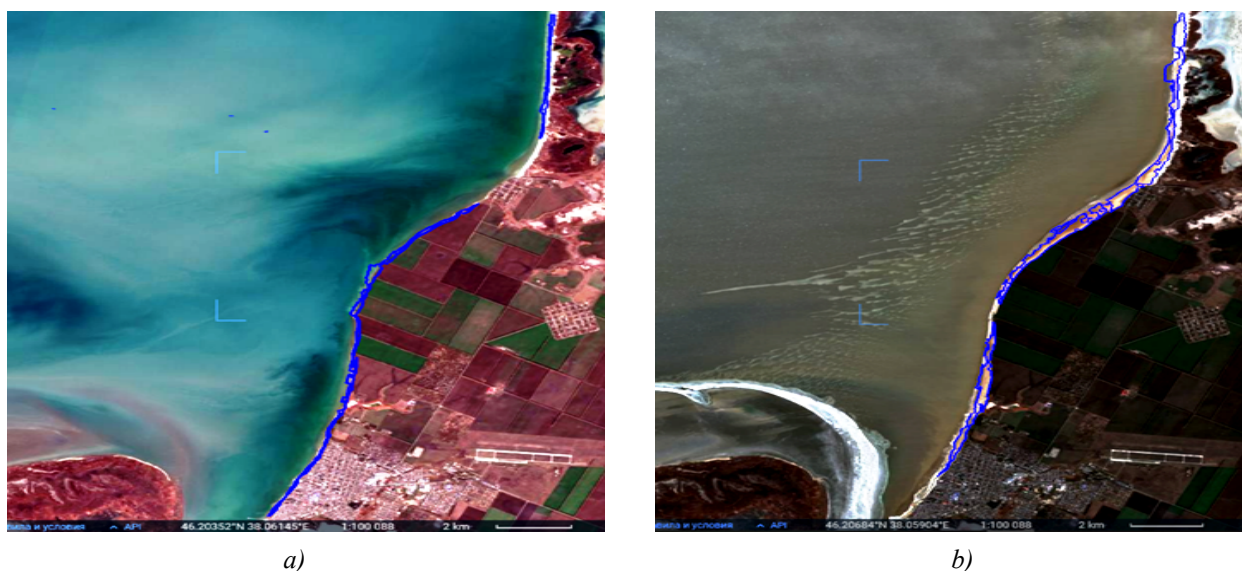


Fig. 12. Snapshots of the study area — Khutor Morozovskiy:
a — November 17, 2019; b — November 22, 2019

Within the framework of numerical experiments, an analysis of snapshots of the study area (Stanitsa Yasenskaya and Khutor Morozovskiy) was conducted for November 17 and 22, 2019. The results demonstrated that the “neural network-LBP” algorithm successfully identifies and tracks changes in the shoreline throughout the examined period. Figures 11 and 12 depict corresponding snapshots showing the dynamics of shoreline changes over the specified period of time.

The results of numerical experiments show the high efficiency of the “neural network-LBP” algorithm for processing boundary contours of aquatic environments. The algorithm enables automatic recognition and input of remote sensing

data into the model, significantly speeding up the modelling process and enhancing its accuracy. In particular, the use of satellite image data for November 17 and 22, 2019, demonstrated the model's ability to accurately forecast predominant wave heights, average wave lengths, and periods of wind waves in the Sea of Azov at different time horizons (initial time, 3, 6, and 9 hours ahead).

Discussion and Conclusions. Modelling of catastrophic surge and seiche events in the Azov Sea is based on solving the system of shallow water wave hydrodynamics equations, which allows for the consideration of complex dynamic processes affecting sea level and wind waves. This study employs the Local Binary Patterns method in combination with neural networks for processing boundary contours of aquatic environments. This approach enables more precise modelling and prediction of extreme sea level fluctuations, which is particularly important for shallow water bodies like the Azov Sea, where such phenomena can have significant economic and environmental consequences. The obtained results underscore the importance and necessity of further research in hydrodynamic modelling and the use of advanced data processing methods to improve forecasts of natural phenomena related to changes in sea level and wind wave dynamics.

References

1. Alekseenko E., Roux B., Sukhinov A., Kotarba R., Fougere D. Coastal hydrodynamics in a windy lagoon. *Nonlinear Processes in Geophysics*. 2013;20(2):189–198. <https://doi.org/10.1016/j.compfluid.2013.02.003>
2. Holthuijsen L.H. *Waves in Oceanic and Coastal Waters*. Cambridge University Press. 2007. <https://doi.org/10.1017/CBO9780511618536>
3. Kirby J.T. (2013). *Advanced Numerical Models for Simulating Tsunami Waves and Runup*. World Scientific Publishing Company. <https://doi.org/10.1142/9101>
4. Komen G.J., Cavaleri L., Donelan M., Hasselmann K., Hasselmann S., Janssen P.A.E.M. (1994). *Dynamics and Modelling of Ocean Waves*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511628955>
5. Mei C.C., Stiassnie M., Yue D.K.P. *Theory and Applications of Ocean Surface Waves: Linear Aspects*. World Scientific Publishing Company. 2005. <https://doi.org/10.1142/5678>
6. The official website of the Unified State Information System on the Situation in the World Ocean. URL: <http://esimo.ru/portal/> (accessed: 11.06.2024)
7. Protsenko S., Sukhinova T. Mathematical modeling of wave processes and transport of bottom materials in coastal water areas taking into account coastal structures. *MATEC Web of Conferences*. 2017;132(2):04002. <https://doi.org/10.1051/mateconf/201713204002>
8. Sukhinov A.I., Chistyakov A.E., Protsenko E.A. Mathematical Modeling of Sediment Transport in the Coastal Zone of Shallow Reservoirs. *Mathematical Models and Computer Simulations*. 2014;6(4):351–363. <https://doi.org/10.1134/S2070048214040097>
9. Panasenkov N.D. Forecasting the Coastal Systems State using Mathematical Modelling Based on Satellite Images. *Computational Mathematics and Information Technologies*. 2023;7(4):54–65. <https://doi.org/10.23947/2587-8999-2023-7-4-54-65>
10. Sukhinov A.I., Protsenko S.V., Panasenkov N.D. Mathematical modeling and ecological design of the marine systems taking into account multi-scale turbulence using remote sensing data. *Computational Mathematics and Information Technologies*. 2022;6(3):104–113. <https://doi.org/10.23947/2587-8999-2022-1-3-104-113>
11. Sukhinov A., Panasenkov N., Simorin A. Algorithms and programs based on neural networks and local binary patterns approaches for monitoring plankton populations in sea systems. *E3S Web of Conferences*. 2022;363:02027. <https://doi.org/10.1051/e3sconf/202236302027>
12. Xu W., Yin X., Zhang W. A Review of Applications of Neural Networks in Coastal and Ocean Engineering. *Ocean Engineering*. 2019;186:106092. <https://doi.org/10.1016/j.oceaneng.2019.106092>
13. Zhang L., Wang L., Yang Z. Deep Learning for Remote Sensing Image Analysis: A Comprehensive Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2018;11(10):3713–3723. <https://doi.org/10.1109/JSTARS.2018.2857630>
14. The official website of Earth observing system. URL: <https://eos.com/landviewer/account/pricing> (accessed: 12.06.2024)
15. The official website of NASA Worldview. URL: <https://worldview.earthdata.nasa.gov> (accessed: 12.06.2024)

Received 06.06.2024

Revised 20.06.2024

Accepted 21.06.2024

About the Authors:

Elena A. Protsenko, Associate Professor of the Department of Mathematics, Leading Researcher, A.P. Chekhov Taganrog Institute (branch) Rostov State University of Economics (48, Initiative St., Taganrog, 347936, RF), Candidate of Physical and Mathematical Sciences, [ORCID](https://orcid.org/0000-0001-9151-1000), eapros@rambler.ru

Natalya D. Panasenko, PhD (Technical Sciences), Senior Lecturer of the Department of Computer Systems and Information Security, Don State Technical University (1, Gagarin Sq., Rostov-on-Don, 344003, RF), [ORCID](#), [ScopusID](#), [ResearcherID](#), [AutorID](#), natalija93_93@mail.ru

Sofia V. Protsenko, Associate Professor of the Department of Mathematics, Researcher, A.P. Chekhov Taganrog Institute (branch) Rostov State University of Economics (48, Initiative St., Taganrog, 347936, RF), Candidate of Physical and Mathematical Sciences, [ORCID](#), rab5555@rambler.ru

Claimed contributor-ship:

All authors have made an equivalent contribution to the preparation of the publication.

Conflict of interest statement

The authors do not have any conflict of interest.

All authors have read and approved the final manuscript.

Поступила в редакцию 06.06.2024

Поступила после рецензирования 20.06.2024

Принята к публикации 21.06.2024

Об авторах:

Проценко Елена Анатольевна, доцент кафедры математики, ведущий научный сотрудник, Таганрогский институт им. А.П. Чехова (филиал) Ростовского государственного экономического университета (347936, РФ, г. Таганрог, ул. Инициативная, 48), кандидат физико-математических наук, [ORCID](#), capros@rambler.ru

Панасенко Наталья Дмитриевна, кандидат технических наук, старший преподаватель кафедры вычислительных систем и информационной безопасности Донской государственной технической университет (РФ, 344003, г. Ростов-на-Дону, пл. Гагарина, 1), [ORCID](#), [ScopusID](#), [ResearcherID](#), [AutorID](#), natalija93_93@mail.ru

Проценко Софья Владимировна, доцент кафедры математики, научный сотрудник, Таганрогский институт им. А.П. Чехова (филиал) Ростовского государственного экономического университета (347936, РФ, г. Таганрог, ул. Инициативная, 48), кандидат физико-математических наук, [ORCID](#), rab5555@rambler.ru

Заявленный вклад соавторов:

Все авторы сделали эквивалентный вклад в подготовку публикации.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Все авторы прочитали и одобрили окончательный вариант рукописи.

MATHEMATICAL MODELLING МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ



Original Theoretical Research



UDC 519.6

<https://doi.org/10.23947/2587-8999-2024-8-2-45-49>

Modelling of Capillary Discharge in Repetition Mode for Short Capillary Systems with Various Filling Methods

Vladimir A. Gasilov^{ID}, Nikita O. Savenko^{ID}✉, Yulia S. Sharova^{ID}

Keldysh Institute of Applied Mathematics of Russian Academy of Sciences, Moscow, Russian Federation

✉ savenkonkt@gmail.com

Abstract

Introduction. Currently, frequency modes of operation of electron accelerators based on capillary discharges are actively investigated. Electrons in these systems are accelerated by femtosecond laser pulses passing through the discharge plasma.

Materials and Methods. The paper presents results of three-dimensional magnetohydrodynamic modelling of the capillary discharge cycle, including stages of filling a short capillary with working gas (hydrogen), formation of the plasma channel, and restoration of the working medium before the start of the next discharge. Calculations were performed assuming the system is under external cooling, which maintains thermal balance at intermediate stages of the working cycle, and under constant conditions of gas supply and evacuation.

Results. The computational experiments demonstrate the capability of generating beams of relativistic electrons with a repetition frequency of approximately one kilohertz.

Discussion and Conclusions. The obtained results allow us to speak about the prospects of using LWFA with a short channel length and a high repetition rate of the capillary discharge.

Keywords: mathematical modelling, magnetohydrodynamics, capillary discharge, laser acceleration of electrons

Acknowledgements. The results were obtained using the equipment of the “Center for Collective Use” IAM RAS (ckp.kiam.ru). The authors are grateful to candidate of physical and mathematical sciences G.A. Bagdasarov for discussing the results of computational experiments.

For citation. Gasilov V.A., Savenko N.O., Sharova Yu.S. Modelling of capillary discharge in repetition mode for short capillary systems with different filling methods. *Computational Mathematics and Information Technologies*. 2024;8(2):45–59. <https://doi.org/10.23947/2587-8999-2024-8-2-45-59>

Оригинальное теоретическое исследование

Моделирование капиллярного разряда в режиме повторения для коротких капиллярных систем при различных способах заполнения

В.А. Гасилов^{ID}, Н.О. Савенко^{ID}✉, Ю.С. Шарова^{ID}

Институт прикладной математики им. М.В. Келдыша Российской академии наук, г. Москва, Российская Федерация

✉ savenkonkt@gmail.com

Аннотация

Введение. В настоящее время активно исследуются частотные режимы работы ускорителей электронов на основе капиллярных разрядов. Электроны в них ускоряются под действием лазерных импульсов фемтосекундного диапазона длительности, пропускаемых через плазму разряда.

Материалы и методы. В работе рассматриваются результаты трехмерного магнитогидродинамического моделирования цикла капиллярного разряда, включающего стадии заполнения короткого капилляра рабочим газом (водород), формирование плазменного канала, восстановление рабочей среды перед началом следующего разряда. Расчеты выполнены в предположении о том, что система находится под внешним охлаждением, которое

обеспечивает температурный баланс на промежуточных этапах рабочего цикла, а также при постоянных условиях подачи и откачки рабочего газа.

Результаты исследования. Результаты вычислительных экспериментов показывают возможность генерации пучков релятивистских электронов с частотой повторения около одного килогерца.

Обсуждение и заключение. Полученные результаты позволяют говорить о перспективности использования КЛПУ с малой длиной канала и высокой частотой повторения капиллярного разряда.

Ключевые слова: математическое моделирование, магнитная гидродинамика, капиллярный разряд, лазерное ускорение электронов

Благодарности. Результаты получены с использованием оборудования ЦКП ИПМ им. М.В. Келдыша РАН (<http://ckp-kiam.ru>). Авторы признательны кандидату физ.-мат. наук Г.А. Багдасарову за обсуждение результатов вычислительных экспериментов.

Для цитирования. Гасилов В.А., Савенко Н.О., Шарова Ю.С. Моделирование капиллярного разряда в режиме повторения для коротких капиллярных систем при различных способах заполнения. *Computational Mathematics and Information Technologies*. 2024;8(2):45–59. <https://doi.org/10.23947/2587-8999-2024-8-2-45-59>

Introduction. The method of generating relativistic electron beams in the field of laser radiation was proposed and substantiated in [1]. Since then, compact laser electron accelerators have found applications in a wide range of fundamental and applied research fields. Among these, developments in free electron lasers (FELs) [2], creation of Compton sources of radiation and electron-positron colliders [3, 4], and others have been highlighted. Works [5–8] have presented theories and results of several experiments on electron beam generation using the Laser Wakefield Acceleration (LWFA) mechanism.

Capillary discharges are widely employed in many experiments in the field of pulsed plasma physics as a simple and convenient tool for generating “quiet”, non-turbulent plasma with reliably controlled parameters. Capillary discharges as a means of creating plasma channels for laser electron acceleration and other applications are extensively discussed in works [7–15].

In several applications (medical-biological research, material sciences, etc.), there is a demand not for high-energy gains achieved by accelerated electrons, as demonstrated in experiment [13], but rather for a high repetition rate (of the order of and exceeding 1 kHz) of electron accelerator pulses. For instance, in [16], a discharge waveguide for LWFA using hydrogen as the working gas was discussed, achieving a repetition frequency of several kHz. Other plasma waveguides for LWFA operating at high repetition rates are described in works [17, 18]. Typically, such experiments utilize relatively short capillaries, approximately 1–3 cm in length. In this regard, several publications highlight the possibility of using short plasma channels to obtain electron beams with energies up to 1 GeV [18–23] as one of the main research outcomes.

This paper presents results from three-dimensional modeling of capillary discharge. The primary focus is on the formation process of the plasma waveguide, which can serve for electron acceleration by a laser pulse. Comparisons are made between computational results and theoretical estimates of accelerated electron energies. The main external factors affecting the capillary in our model include the gas flow into the capillary feeding channels and the plasma heat exchange with the capillary walls.

The research multifunctional code MARPLE, developed at the Institute of Applied Mathematics named after M.V. Keldysh [20], is utilized for modeling the described process.

Materials and Methods. Computational experiments to study the dynamics of capillary discharge plasma are conducted using a single-fluid two-temperature magnetohydrodynamic (MHD) model. This model incorporates electron-ion energy exchange, thermal conductivity of the ion and electron components of the plasma. The main system of equations solved includes [24]:

Continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{V}) = 0,$$

momentum balance equation

$$\begin{aligned} \rho \left(\frac{\partial}{\partial t} + \mathbf{V} \cdot \nabla \right) \mathbf{V} &= -\nabla(p_i + p_e) + \frac{1}{c} \mathbf{j} \times \mathbf{B}, \\ \rho \left(\frac{\partial}{\partial t} + \mathbf{V} \cdot \nabla \right) \varepsilon_i + p_i \nabla \cdot \mathbf{V} &= \nabla(\kappa_i \nabla T_i) - Q_{ie}, \\ \rho \left(\frac{\partial}{\partial t} + \mathbf{V} \cdot \nabla \right) \varepsilon_e + p_e \nabla \cdot \mathbf{V} &= \nabla(\kappa_e \nabla T_e) + \frac{\mathbf{j}^2}{\sigma} + Q_{ie} - Q_{\text{rad}}, \end{aligned}$$

Maxwell's equations (excluding displacement current)

$$\mathbf{j} = \frac{c}{4\pi}(\nabla \times \mathbf{B}), \quad \frac{\partial \mathbf{B}}{\partial t} = -c(\nabla \times \mathbf{E}), \quad \Delta \cdot \mathbf{B} = 0.$$

The main system of equations is closed by the equation of state in the form of

$$p_i = p_i(\rho, \epsilon_i), \quad p_e = p_e(n_e, \epsilon_e),$$

and Ohm's law in the form of $\mathbf{E} = \frac{1}{c}(\mathbf{V} \times \mathbf{B}) + \frac{\mathbf{j}}{\sigma}$.

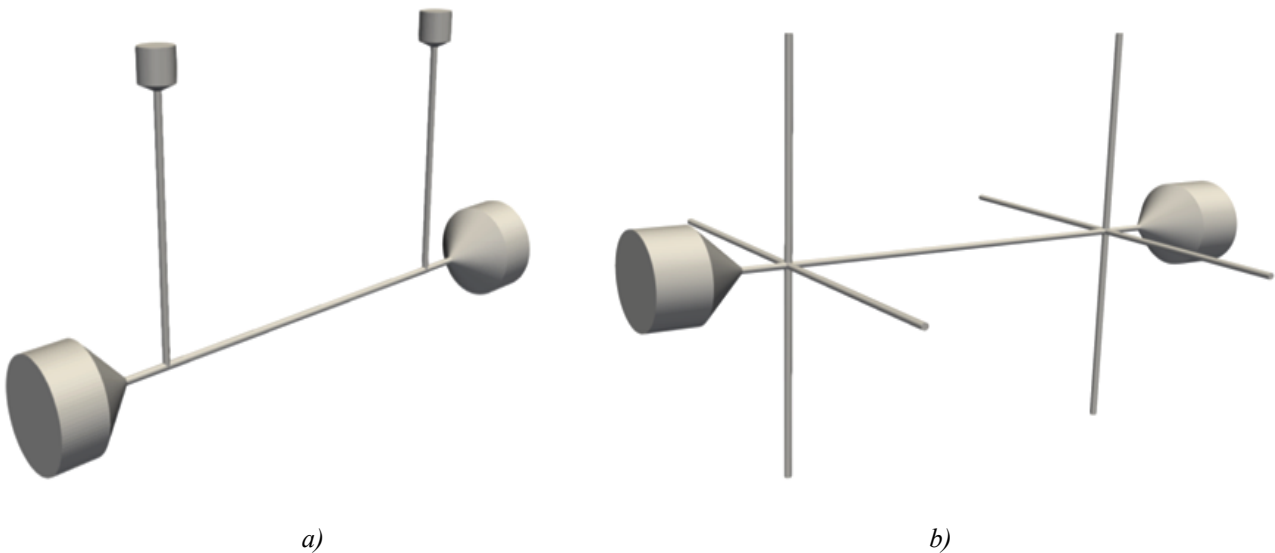
It is assumed that throughout the entire duration of the electric discharge, the condition of quasi-neutrality of the gas-plasma medium is maintained $n_e - Zn_i = 0$.

The operational cycle of the capillary is divided into three consecutive stages. The first stage involves filling the capillary with cold gas (hydrogen) until flow stabilization occurs. The second stage is the actual electric discharge necessary for forming the discharge channel in a fully ionized environment, where the termination of the electrical pulse is characterized by rapid recombination within the discharge plasma. In the final, third stage of the cycle, hot gas escapes from the capillary and is replaced by a fresh portion of gas from the feeding channels. The stages of filling and purging are calculated in a single-temperature approximation, as the presence of the electron component during this period is negligible.

The study considers two geometrical configurations of the system: with two and with eight feeding channels. The results of the calculations are compared with those from [21], obtained under the same conditions of capillary discharge implementation but in a simpler problem setup, considered in a highly idealized two-dimensional cylindrical (r, z) geometry. The capillary of circular cross-section is assumed to have a length of 2 cm and a diameter of 300 μm (see Fig. 1). The computational domain is structured as follows: the main capillary channel is modeled as a tube with open ends, to which the feeding channels of the same diameter as the capillary (300 μm) are attached at a distance of 0.8 cm from the center of the main tube. Electrodes with a half-aperture angle of 45° and a length of 0.2 cm are positioned at the ends of the capillary, followed by a free gas exit area with a length of 0.2 cm.

The construction of the computational domain for the capillary with eight feeding channels is carried out by assuming that the number of feeding channels per half-length of the capillary is four, cross-attached to the capillary. This geometry of the entire structure allows for a more homogeneous gas flow in the region where the feeding channels connect with the capillary.

The computational mesh is built considering the symmetry of the geometrical model of the system with eight feeding channels, thereby limiting calculations to one eighth of the entire area filled with the working substance (feeding channels and the main capillary tract). The electromagnetic field is computed for the capillary channel in a dielectric, as commonly implemented in most designs [2, 16, 21]. High-conductivity metal electrodes (copper, aluminum) are attached to the ends of the capillary in accordance with this design. Consequently, during the calculation stage of the electric discharge, boundary conditions for the equations of electrodynamics are applied from the general magnetohydrodynamic system. Gas dynamic equations are solved using standard impermeable wall conditions for the capillary walls, while at the open ends of the capillary, gas parameters are computed according to the methodology for implementing non-reflecting boundary conditions [22].



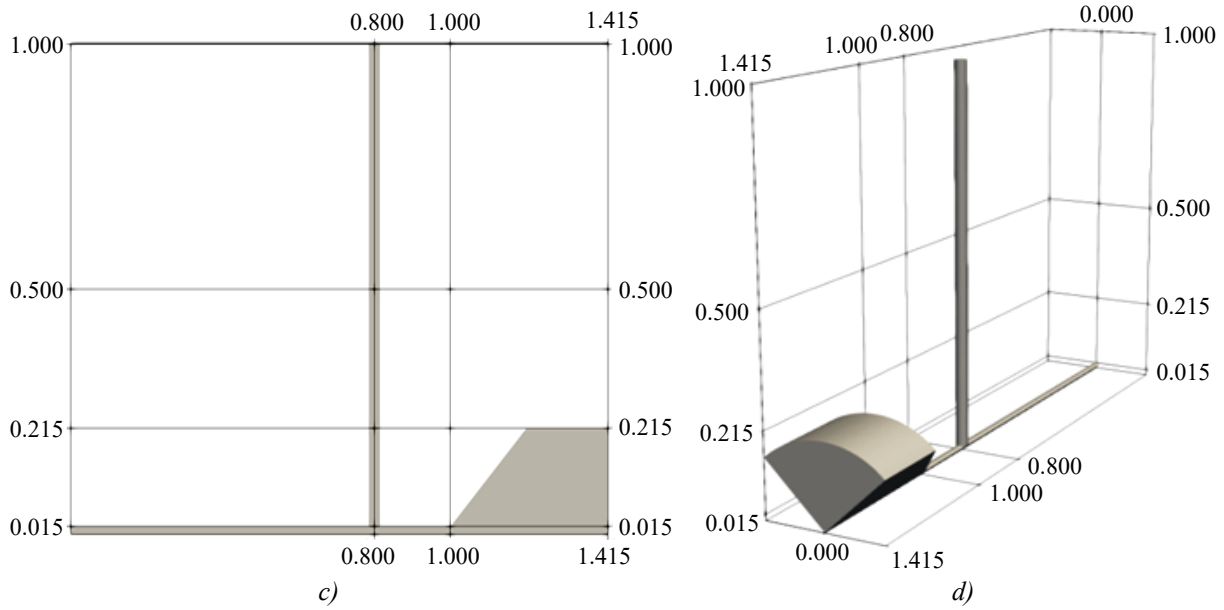


Fig. 1. Geometry of the computational domain: *a* — general view for the system with two feeding channels; *b* — general view for the system with eight feeding channels; *c* — characteristic dimensions of the domain are identical for both considered variants of the capillary system construction; *d* — characteristic dimensions of the domain for the system with eight feeding tubes

Calculations were performed for two consecutive operating cycles. During the capillary filling stage, both primary and repeated, the inlet pressure into the feeding channels for the two-channel system was set to $p = 0.125$ bar with a temperature of 300 K, while for the eight-channel configuration, $p \approx 0.083$ bar at 300 K was used. These inlet gas parameter values ensure equal gas flow rates during system filling and approximately equal values of gas flow parameters along the capillary axis after reaching steady-state flow conditions. Once steady-state flow is achieved within the system, the filling stage is considered complete. The gas parameters attained in the capillary at this stage serve as initial data for the discharge phase.

In accordance with numerous experiments [16–19, 21], the following dependence for the electrical current through the capillary was used for the calculations:

$$I = I_0 \frac{t}{t_{\max}} \exp \left(1 - \frac{t}{t_{\max}} \right), \quad (1)$$

where $I_0 = 210$ A; t is the current time from the start of the discharge; $t_{\max} = 0.15$ μ s is the time of maximum current strength. The current pulse profile is shown in Fig. 2.

The azimuthal component of the magnetic field at the insulator boundary, specified as a boundary condition for the magnetic field, is approximately taken as $B_\phi = 2I(t)/R$, where $I(t)$ is calculated using equation (1), and R is the radius of the capillary tube. The second stage of the capillary operation concludes when the discharge current ceases. Over the subsequent short interval, the ionized substance electro-conducted recombines. The gas state during this period is assumed as the initial state for the subsequent refilling stage.

During the final stage, the process of restoring the distribution of neutral hydrogen in the capillary is simulated. This includes the expulsion of hot hydrogen from the capillary and its replacement with a new portion of cold hydrogen from the feeding channels, initiating the start of the refilling process.

For the case of eight feeding tubes, calculations for all three discharge stages are performed within one eighth of the total flow area. This is feasible due to the symmetry of the capillary construction with feeding tubes. The electrode at the end of the capillary is considered as a cylindrical layer of metal — a “ring” approximately one-tenth of the radius thick.

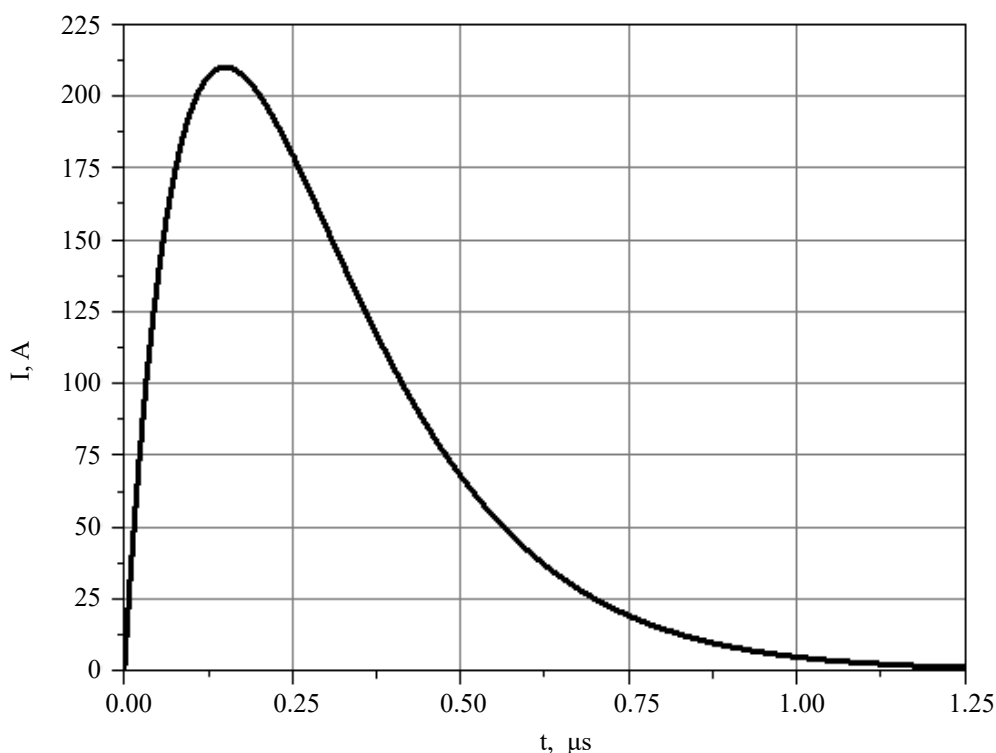


Fig. 2. Current profile as a function of time

Results

Primary filling. The primary filling stage concludes upon the establishment of steady hydrogen flow through the capillary. Fig. 3 and 4 depict density profiles along the capillary axis at various time points for systems with two and eight feeding channels (Fig. 3), as well as in the two-dimensional model (Fig. 4) with simplified (ring-shaped) gas delivery into the capillary. The corresponding stages of gas flow evolution are illustrated.

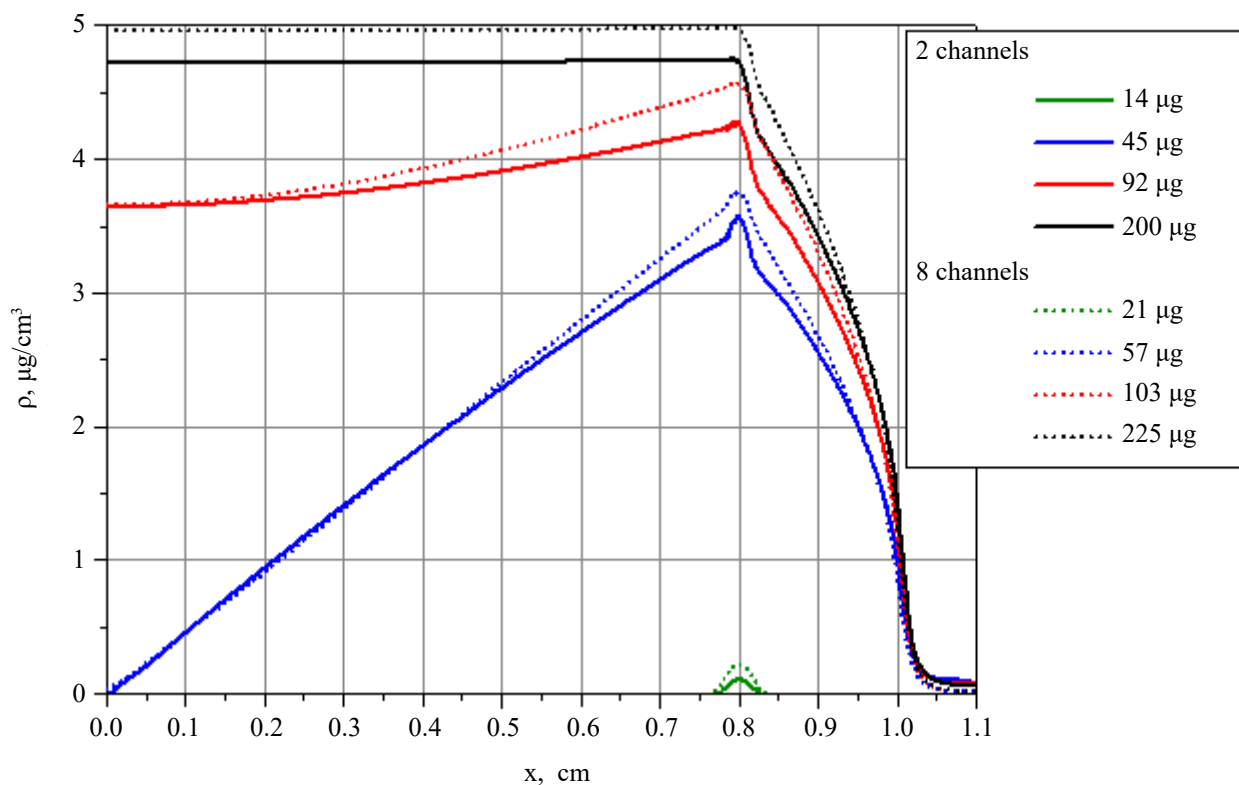


Fig. 3. Density distribution along the axis of the capillary for two geometries at corresponding time points

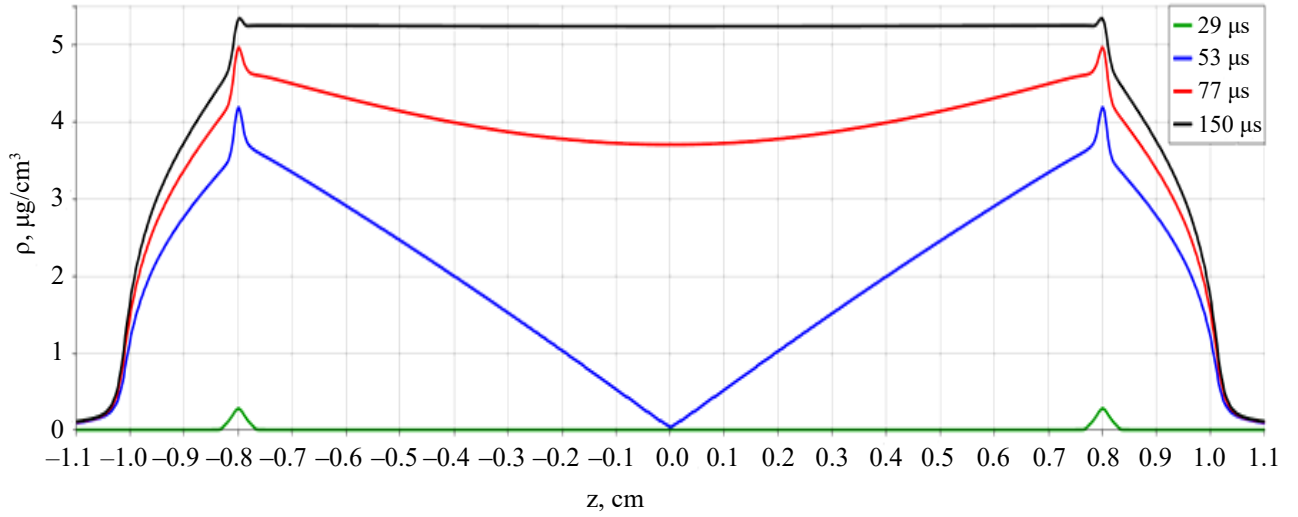


Fig. 4. Density distribution along the axis of the capillary for the two-dimensional model discussed in [21]

The slight difference in parameters upon reaching steady state does not significantly influence subsequent stages. As demonstrated in Figure 5, for three different feeding channel geometries, the establishment times are as follows: 2 feeding channels — 200 μs , 8 feeding channels — 225 μs , two-dimensional model of capillary feeding — 125 μs .

Due to thermal conductivity, the gas temperature at this stage equalizes and matches the temperature of the capillary wall, which is 300 K.

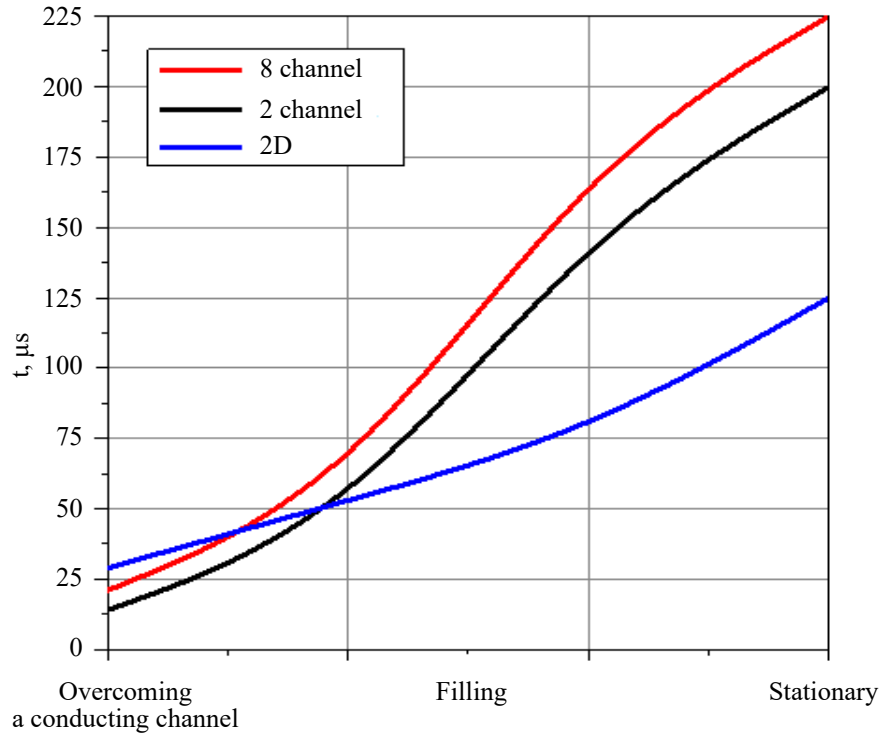


Fig. 5. Establishment time of flow during the primary filling stage for different geometries

Fig. 6 illustrates the distribution of mass flow (g/s) along the system and across the main channel near the inlet of the feeding tube (0.82 cm) for eight and two feeding tubes.

The slight yet noticeable asymmetry in flow parameters observed in different cross-sections in the case of two feeding tubes filling the capillary is indicative of the quality of the waveguide channel, which must possess sufficiently high radial symmetry to ensure the high efficiency of electron acceleration by a laser pulse. The symmetry of flow is particularly crucial in the vicinity of the capillary axis, where the channel with parameters necessary for electron acceleration is formed.

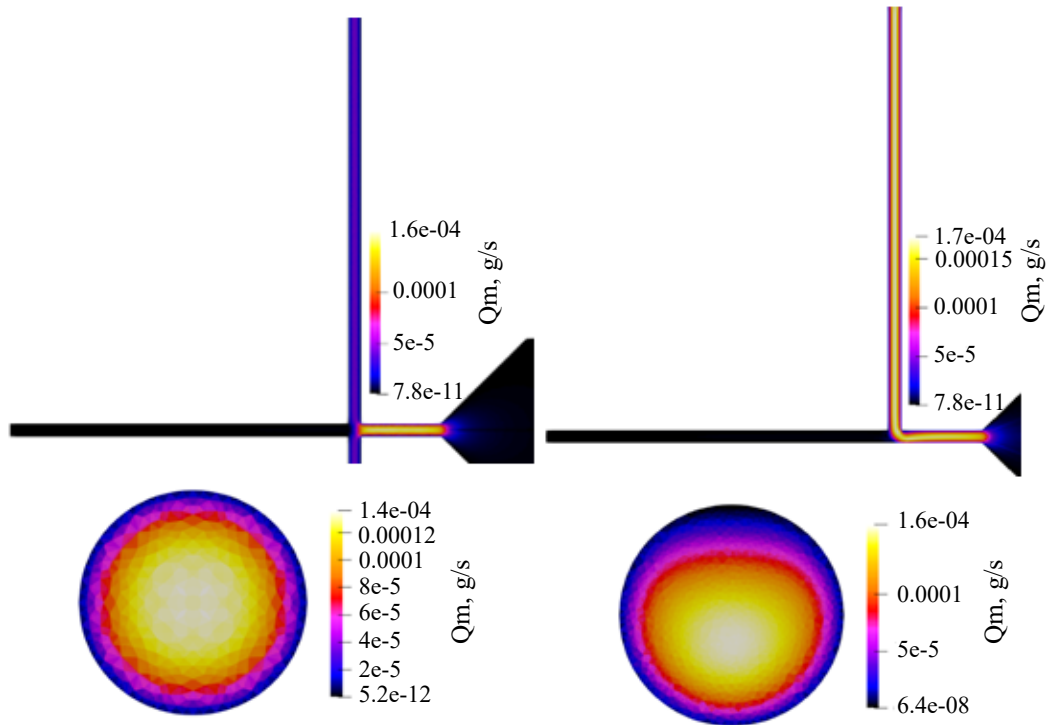


Fig. 6. Mass flow distribution in the cross-section of the capillary near the inlet of the feeding channel for eight and two feeding channels

Discharge. To form a plasma channel capable of efficiently guiding an accelerating laser pulse through it, an electrical discharge is applied to the cold gas flow. For modeling the discharge, we utilize the initial state of the gas obtained after establishing it during the filling stage.

The MHD model we employ does not include descriptions of the electrical breakdown stage. The breakdown lasts about 10 ns and minimally affects the properties of the plasma channel. For modeling the second stage, the gas is transitioned into a weakly ionized state by increasing the temperature to 0.3 eV. Fig. 7 and 8 demonstrate that this artificially induced initial temperature jump constitutes less than 5 % of the maximum discharge temperature and does not significantly impact the state of the electro-discharge plasma.

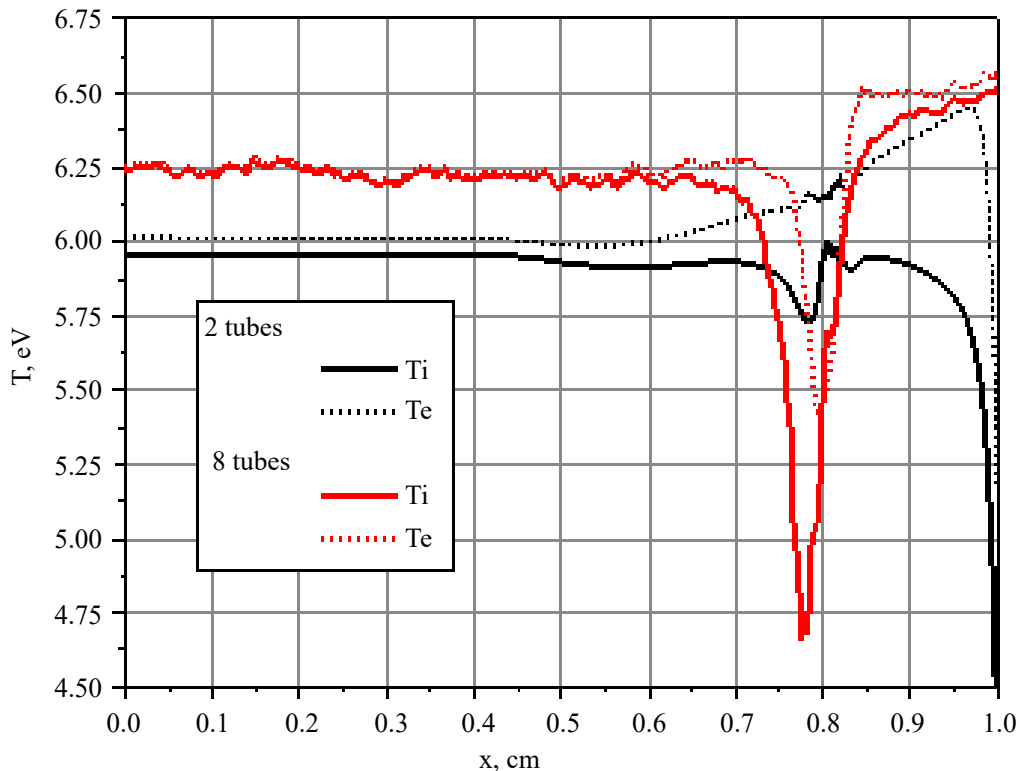


Fig. 7. Distribution of electron and ion temperature components along the axis of the capillary at the moment of maximum current for two and eight feeding channels

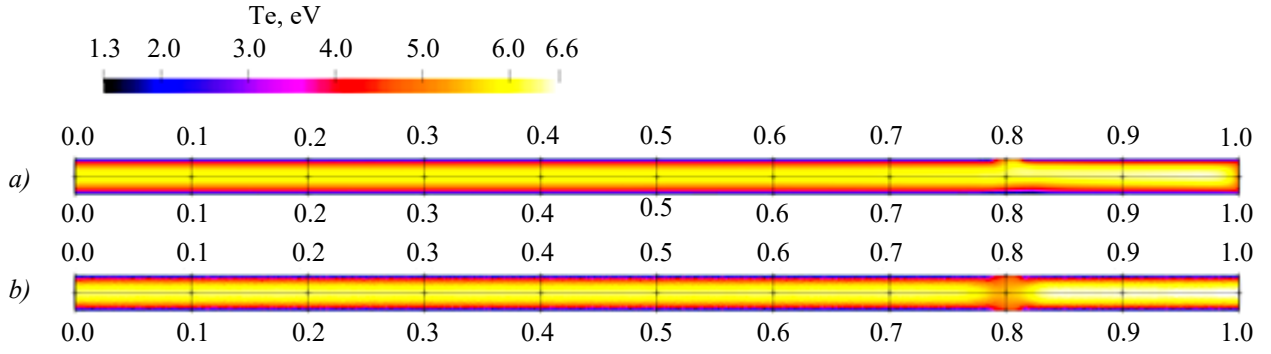


Fig. 8. Distribution of electron temperature in the longitudinal section at the moment of maximum current:
 a — 8 feeding channels; b — 2 feeding channels

The following distribution of electron density along the axis of the capillary for various time points has been obtained (Fig. 9). The slight increase in density near the exit during the initial period is due to the capillary gas experiencing some retardation as it displaces the gas that filled the nozzle towards the end of the first stage of the filling process.

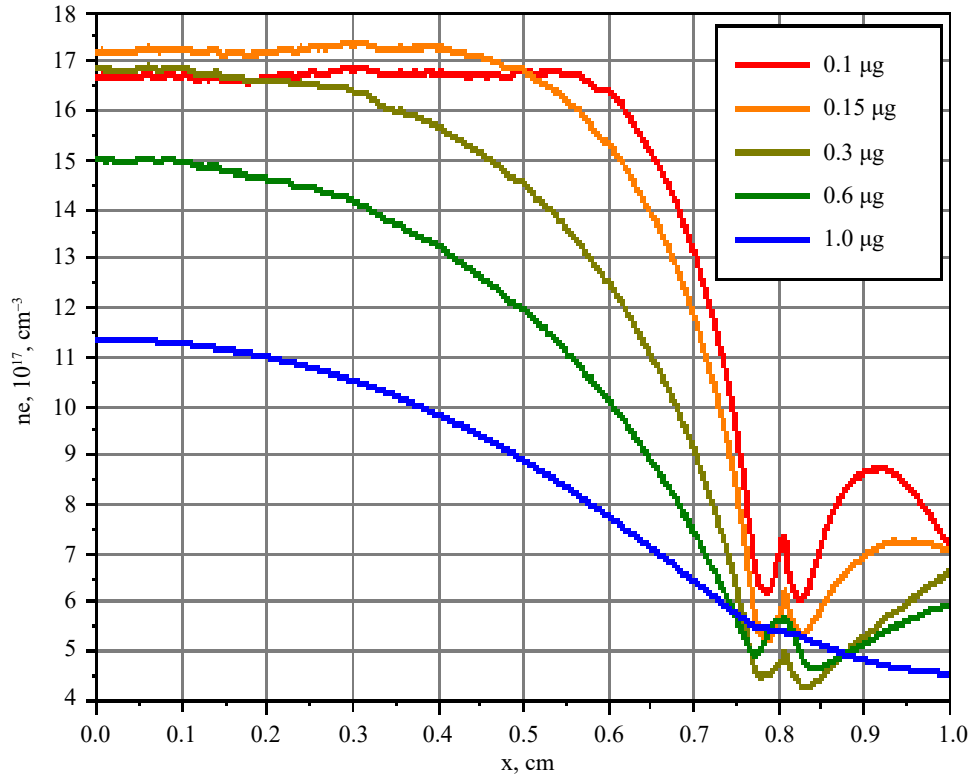


Fig. 9. Electron concentration along the axis of the capillary at different times relative to the start of the electrical discharge for a system with eight feeding channels

For cases with two and eight feeding channels, the distribution of plasma density during discharge is nearly identical with minor differences in the region from 0.75 to 1 cm (Fig. 10). This is caused by the displacement of discharge plasma into the feeding channels during the discharge process; it should be noted that the cross-sectional area through which gas enters the capillary is four times larger in the case of eight tubes compared to two tubes.

The most pronounced non-uniformity is observed in the flow section near the feeding channels, as shown in Fig. 11. In the junction area of the feeding channels, the electron density is approximately half that of the central part of the capillary. Additionally, good symmetry of distribution can be noted. It is reiterated that radial symmetry in the distribution and homogeneity of the channel along its axis are crucial for acceleration. The significant gradient of density along the boundaries of the forming waveguide also plays a crucial role in creating favorable conditions for the collimation of laser radiation in the waveguide, as clearly demonstrated in Figure 15.

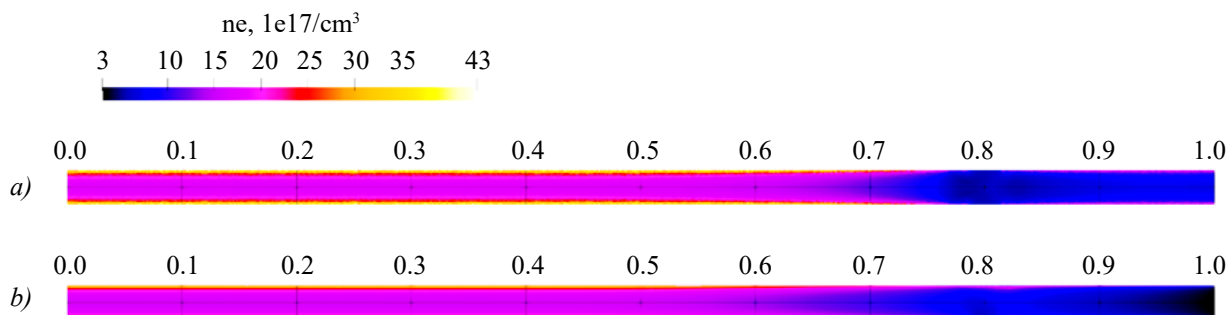


Fig. 10. Distribution of electron density in the longitudinal section at the moment of maximum current:
a — 8 feeding channels; b — 2 feeding channels

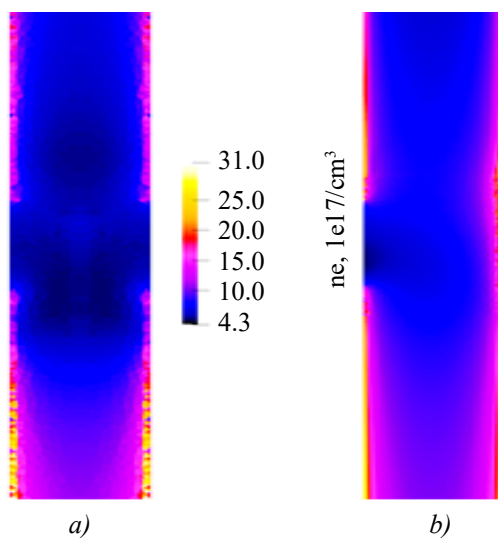


Fig. 11. Distribution of electron density in the longitudinal section near the attachment of feeding channels at the moment of maximum current: a — 8 feeding channels; b — 2 feeding channels

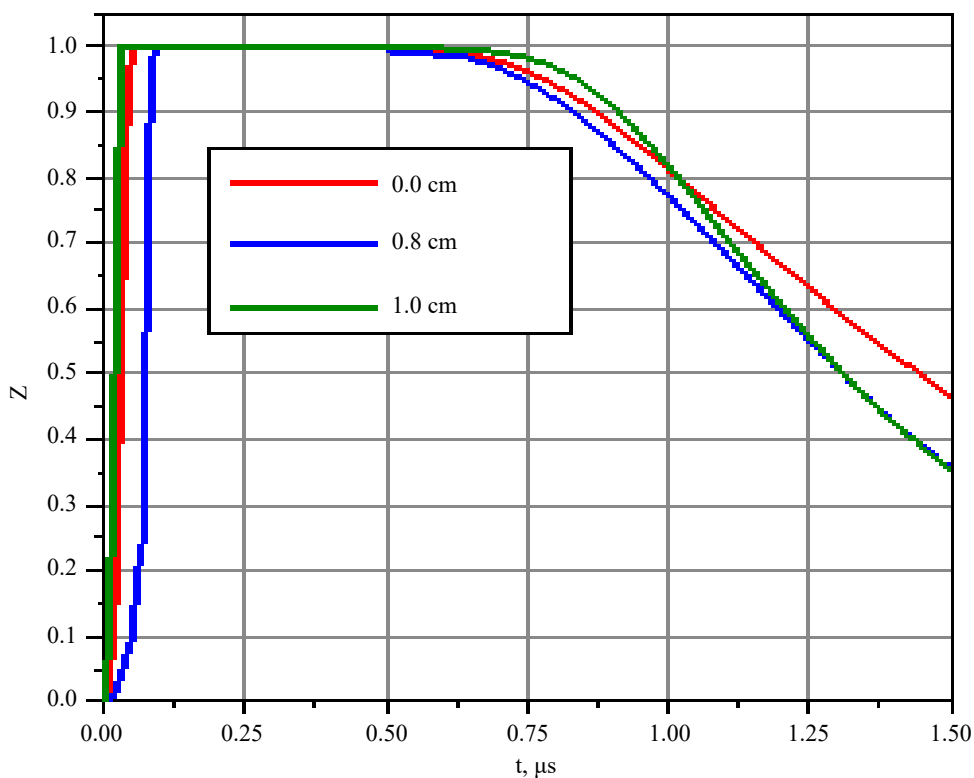


Fig. 12. Hydrogen ionization levels at three control points along the axis of the capillary
(distance from the center indicated) as a function of time

The channel is considered prepared to pass an accelerating laser pulse through it when the hydrogen ionization level reaches unity. We examine the ionization evolution (Fig. 12) at three control points along the axis taken at specific distances from the capillary center: 0 cm — center of the main channel, 0.8 cm — point opposite the feeding channels, 1 cm — end of the capillary. Based on the data presented in Figure 12, it can be concluded that a waveguide with the required properties forms along the entire length of the capillary and exists from the moment of maximum current (150 ns) to approximately 700 ns, i. e., for a period of approximately 500–600 ns.

From the density distribution presented in Figure 10, it follows that there are differences in the plasma waveguide structures depending on the geometry of the feeding channels. This difference is insignificant for the electron density distribution in the central part of the capillary. Discrepancies in electron density distribution near the feeding channels and further towards the open end are much more pronounced — up to 50 % lower than the average density in the segment from 0.75 to 1 cm along the capillary length. In the case of eight feeding channels, a significant portion of the plasma is displaced into them, resulting in a region of reduced plasma density near the entrance of the feeding channels. In the case of two channels, plasma displacement into the channels is less noticeable, and plasma flows more through the ends of the capillary.

The most important characteristic of the plasma concerning its use as a working medium for electron acceleration by a laser pulse is the electron concentration along the axis of the capillary. The average electron concentration value at the moment of maximum current is nearly identical for both cases in the flow region between the central section of the capillary and up to a distance of approximately 0.75 cm from the center, with slight differences in the end regions, and is approximately $n_e \approx 1.8 \times 10^{18} \text{ cm}^{-3}$. In the region of maximum rarefaction, from 0.75 to 1 cm, the electron concentration is approximately $n_e \approx 0.8 \times 10^{18} \text{ cm}^{-3}$. It should be noted that these concentration values are in good agreement with the concentration value $\sim 2 \times 10^{18} \text{ cm}^{-3}$, determined in [2, 21] for a capillary with the same diameter and length as in the present work, calculated for an experiment with a laser pulse energy of $\sim 3 \text{ J}$ with a duration of about 30 fs.

Fig. 13 and 14 show the transverse distribution of density and electron concentration at the center of the tube ($x = 0 \text{ cm}$). The obtained density values for eight and two feeding tubes are close to the values obtained in the two-dimensional variant. Minor differences in electron density distributions for the two capillary filling options are due to the different mesh structures used in these variants, which are also reflected in the temperature profile (Fig. 7, 8). Figure 15 presents the electron concentration distribution across the capillary at various distances from the center.

The transverse dynamics of capillary plasma, leading to the formation of a plasma channel suitable for laser acceleration, is analogous to that considered in [9], where calculations were based on a one-dimensional flow model under cylindrical symmetry approximation. Three-dimensional modeling shows that the lifetime of the plasma channel is determined by the plasma flow into the open ends of the capillary and partially backflow into the feeding channels.

The duration of the electrical pulse has less influence on the channel lifetime compared to a longer capillary — 10 centimeters or more.

Figures 16 and 17 show the distribution of electron concentration at the location of the attachment of feeding channels. In the case of eight channels, this area exhibits greater optical transparency compared to the two-channel variant. Apart from lower transparency, the “two-channel” system notably disrupts the radial symmetry of the plasma channel. The duration of the electrical discharge stage is measured from the moment of discharge initiation until nearly complete hydrogen recombination.

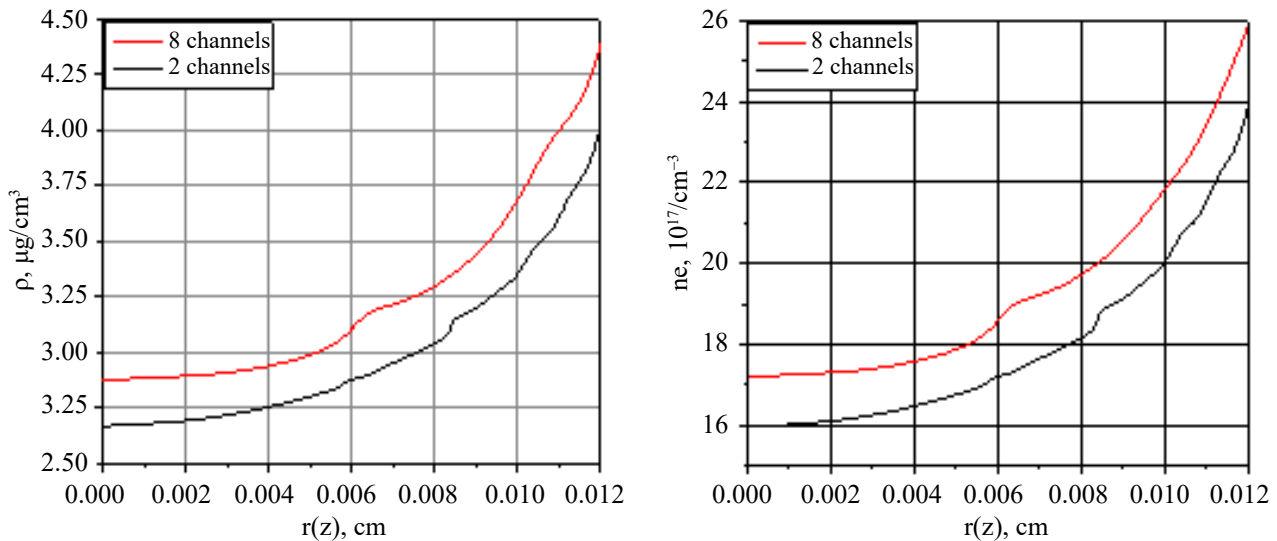


Fig. 13. Transverse distribution of electron density and concentration at the center of the capillary for two and eight feeding channels at the moment of maximum current

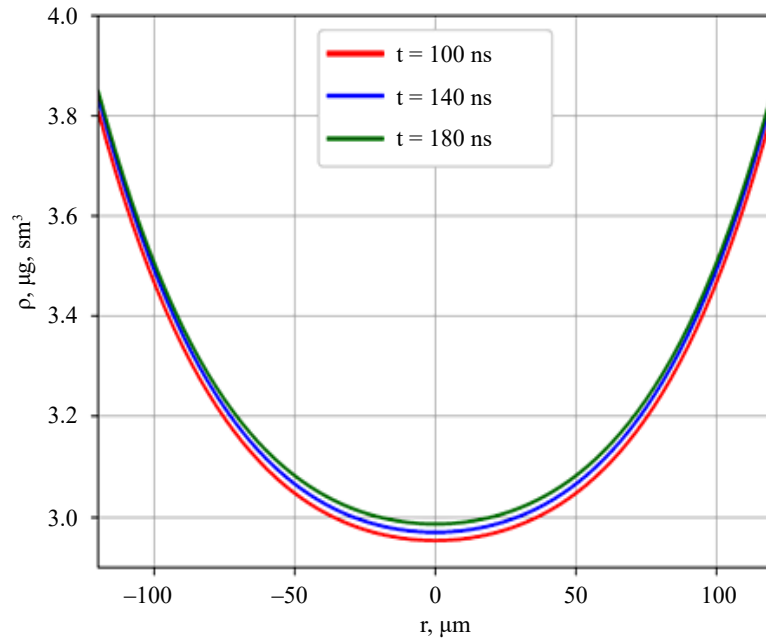


Fig. 14. Transverse distribution of electron density at the center of the capillary in the two-dimensional model [21]

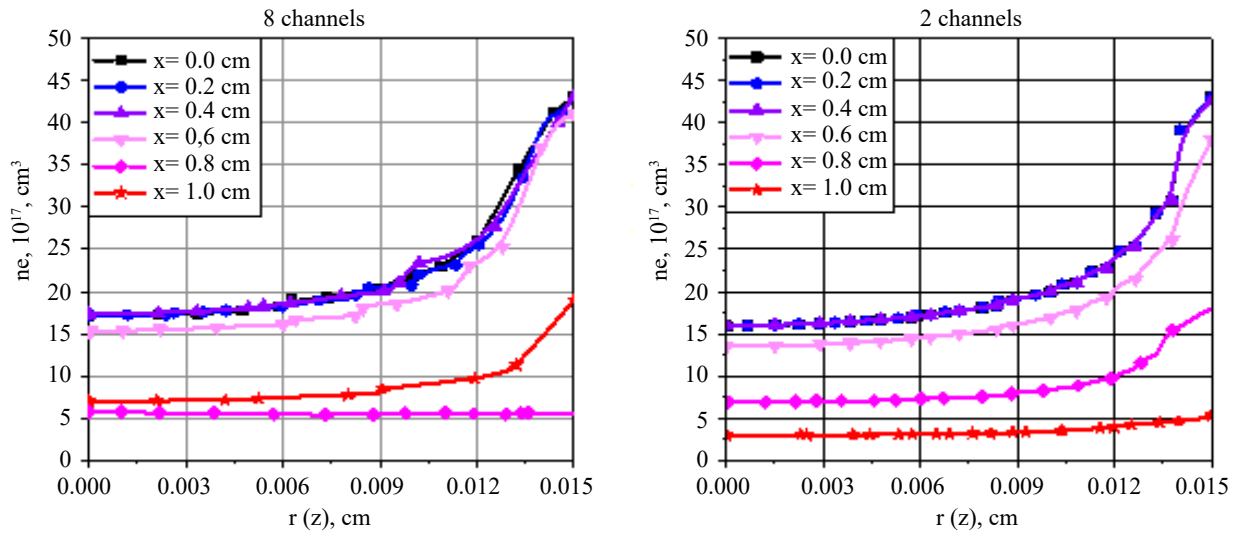
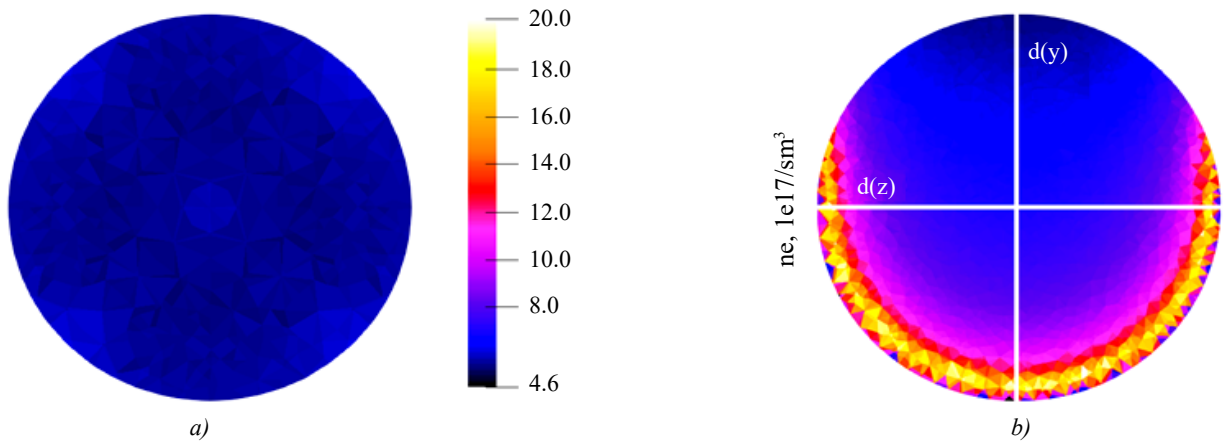


Fig. 15. Transverse distribution of electron concentration at various distances from the center of the capillary for eight and two feeding channels


 Fig. 16. Electron concentration distribution across the capillary section near the attachment of feeding channels ($x = 0.8$):
 a — 8 feeding channels; b — 2 feeding channels

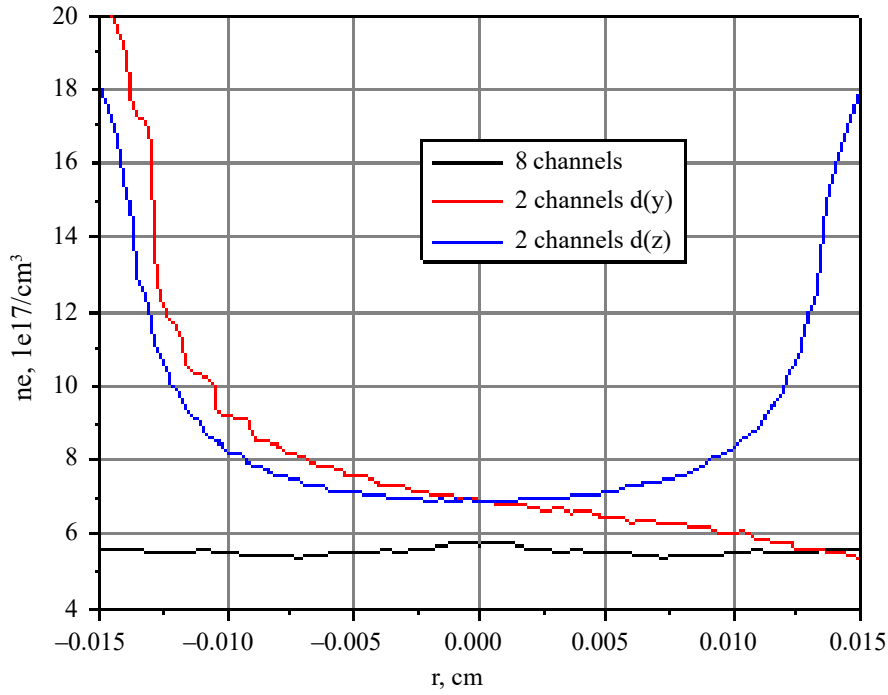


Fig. 17. Distribution of electron concentration across the diameter of the section in the case of eight feeding channels, as well as along the vertical and horizontal axes in the case of two feeding channels

Gas Exhaustion and Capillary Refilling. Upon completion of hydrogen recombination within the capillary, a sufficient amount of hot gas remains under high pressure. Active gas expansion continues towards the open ends and feeding channels, starting from the initiation of the electrical discharge.

Gas exhaustion is considered complete when the hydrogen pressure in the feeding reservoir exceeds that of the expanding gas, initiating the process of refilling the capillary with cold gas. It is noteworthy that during refilling, the central part of the capillary already contains gas which, as it cools down, returns to its initial state.

The refilling process occurs almost identically to the initial filling. The difference lies in a denser environment compared to its initial state, as well as a higher gas temperature within the “capillary + feeding channels” system. The times for initial and repeat fillings are found to be close; for the sake of analysis convenience, they are considered equal in the capillary operation cycles.

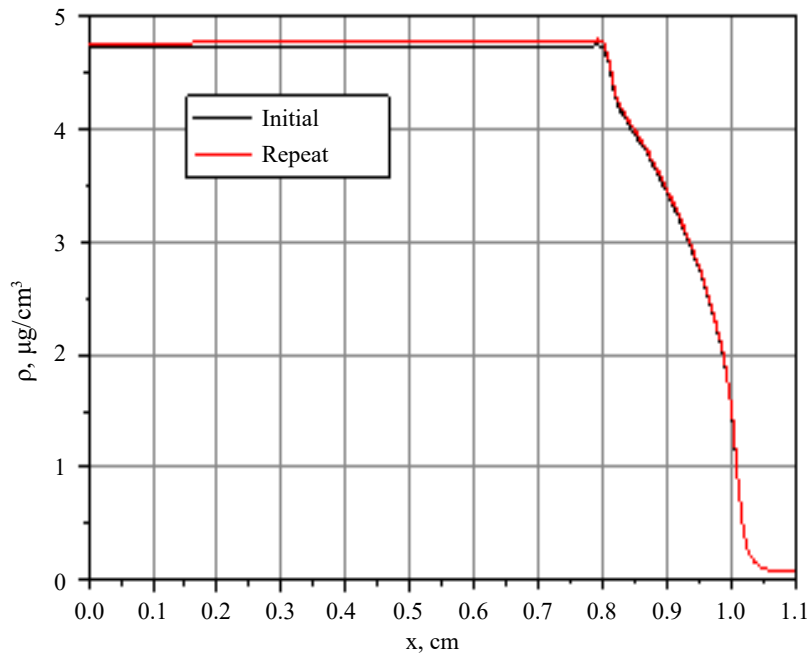


Fig. 18. Density profiles along the axis of the capillary for initial and repeat filling scenarios, illustrated for the case with two feeding channels

Fig. 18 shows a comparison of density profiles along the axis for initial and repeat filling for the case with two feeding channels.

Discussion and Conclusion. The calculated time for one cycle of “filling — discharge — relaxation” is presented in Table 1. In the calculation based on the two-dimensional model, the time for repeat filling does not exceed half of the time for initial filling.

Table 1

Calculated time for one cycle of “filling — discharge — relaxation”

Mounting Type	8 channels	2 channels	2D
Filling time, μs	225	200	125
Ionized state, μs	5	5	5
Release of heated gas, μs	< 50	< 45	
Total cycle time, μs	$\lesssim 280$	$\lesssim 250$	$\lesssim 130$ (перв.) $\lesssim 60$ (повт.)

The shape of the feeding channels (2 tubes, 8 tubes, two-dimensional filling model) has the greatest influence on the operating frequency of the short capillary. The diameters and lengths of the feeding channels evidently affect the filling rate. To a lesser extent, the temperature and pressure of the gas entering the system affect the cycle duration. The intensity of filling is determined by experimental conditions [2].

The mathematical model of the capillary system developed by the authors allows concluding that the capillary can operate in the frequency range from 1 to 16.5 kHz.

In conclusion, let's assess the energy gained by electrons during their acceleration by a laser pulse in the capillary discharge channel, for which we can use, for example, the formula proposed in [23]:

$$W \text{ (GeV)} \sim \frac{0,4 I (W/cm^2)}{n_e (1/cm^3)}, \quad (2)$$

where W is the energy of accelerated electrons; $I \leq 10^{18} W/cm^2$ is the intensity of the accelerator laser; n_e is the average electron density along the path of the beam.

As initial data for estimation, we take the parameters of the plasma medium at the center of the capillary close to the maximum current strength (~150 ns). Using the data obtained during the discharge phase, we can expect electron energies during acceleration, according to formula (2), $W_e \leq 0.5$ GeV, which holds true for all three considered geometrical variants.

The obtained values of acceleration and potential operating frequency of the accelerator indicate the prospect of using a short-length capillary plasma accelerator. An important characteristic of the electron beam is its emittance. For a precise assessment, modeling based on a Particle-in-Cell approach, for example, in a ponderomotive approximation [5], is necessary. Corresponding computational experiments are the subject of further research.

References

1. Tajima T., Dawson J.M. Laser Electron Accelerator. *Phys. Rev. Lett.* 1979;43:267. <https://doi.org/10.1103/PhysRevLett.43.267>
2. Molodozhentsev A., Korn G., Maier A., Pribyl L. LWFA-driven Free Electron Laser for ELI-Beamlines. *ICFA Advanced Beam Dynamics Workshop on Future Light Sources*. JACoW, Geneva. 2018;60:62–67. <https://doi.org/10.18429/JACoW-FLS2018-TUA2WC02>
3. Leemans W., Esarey E. Laser-driven plasma-wave electron accelerators. *Phys. Today*. 2009;62(3):44.
4. Schroeder C.B., Esarey E., Geddes C.G.R., Benedetti C., Leemans W.P. Physics considerations for laser-plasma linear colliders. *Phys. Rev. Accel. Beams*. 2010;13:101301. <https://doi.org/10.1103/PhysRevSTAB.13.101301>
5. Leemans W.P., Gonsalves A.J., Mao H.-S., Nakamura K., Benedetti C., Schroeder C.B., et al. Multi-GeV Electron Beams from Capillary-Discharge-Guided Subpetawatt Laser Pulses in the Self-Trapping Regime. *Phys. Rev. Lett.* 2014;113:245002. <https://doi.org/10.1103/PhysRevLett.113.245002>
6. Gonsalves A.J., Nakamura K., Daniels J., Benedetti C., Pieronek C., de Raadt T.C.H., et al. Petawatt laser guiding and electron beam acceleration to 8 GeV in a laser-heated capillary discharge waveguide. *Phys. Rev. Lett.* 2019;122:084801. <https://doi.org/10.1103/PhysRevLett.122.084801>
7. Esarey E., Schroeder C.B., Leemans W.P. Physics of laser-driven plasma-based electron accelerators. *Rev. Mod. Phys.* 2009;81:1229. <https://doi.org/10.1103/RevModPhys.81.1229>
8. Spence D.J., Hooker S.M. Investigation of a hydrogen plasma waveguide. *Phys. Rev. E*. 2001;63:015401(R). <https://doi.org/10.1103/PhysRevE.63.015401>

9. Bobrova N.A., Esaulov A.A., Sakai J.-I., Sasorov P.V., Spence D.J., Butler A., et al. Simulations of a hydrogen-filled capillary discharge waveguide. *Phys. Rev. E*. 2002;65:016407. <https://doi.org/10.1103/PhysRevE.65.016407>
10. Hosokai T., Kando M., Dewa H., Kotaki H., Kondo S., Hasegawa N., et al. Optical guidance of terrawatt laser pulses by the implosion phase of a fast z-pinch discharge in a gas-filled capillary. *Opt. Lett.* 2000;25:10–12. <https://doi.org/10.1364/OL.25.000010>
11. Kameshima T., Kotaki H., Kando M., Daito I., Kawase K., Fukuda Y., et al. Laser pulse guiding and electron acceleration in the ablative capillary discharge plasma. *Phys. Plasmas*. 2009;16:093101. <https://doi.org/10.1063/1.3212589>
12. Gonsalves A.J., Nakamura K., Lin C., Panasenkov D., Shiraishi S., Sokollik T, et al. Tunable laser plasma accelerator based on longitudinal density tailoring. *Nat. Phys.* 2011;7:862–866. <https://doi.org/10.1038/nphys2071>
13. Pieronek C., Gonsalves A., Benedetti C., Bulanov S., van Tilborg J., Bin J., et al. Laser-heated capillary discharge waveguides as tunable structures for laser-plasma acceleration. *Phys. Plasmas*. 2020;27:093101. <https://doi.org/10.1063/5.0014961>
14. Bobrova N.A., Sasorov P.V., Benedetti C., Bulanov S.S., Geddes C.G.R., Schroeder C.B., et al. Laser-heater assisted plasma channel formation in capillary discharge waveguides. *Phys. Plasmas*. 2013;20:020703. <https://doi.org/10.1063/1.4793447>
15. Bagdasarov G.A., Bobrova N.A., Olkhovskaya O.G., Gasilov V.A., Benedetti C., Bulanov S.S., et al. Creation of axially uniform plasma channel in laser-assisted capillary discharge. *Phys. Plasmas*. 2021;28:053104. <https://doi.org/10.1063/5.0046428>
16. Gonsalves A.J., Liu F., Bobrova N.A., Sasorov P.V., Pieronek C., Daniels J, et al. Demonstration of a High Repetition Rate Capillary Discharge Waveguide. *J. Appl. Phys.* 2016;119:033302. <https://doi.org/10.1063/1.4940121>
17. Alejo A., Cowley J., Picksley A., Walczak R., Hooker S.M. Demonstration of kilohertz operation of hydrodynamic optical-field-ionized plasma channels. *Phys. Rev. Accel. Beams*. 2022;25:011301. <https://doi.org/10.1103/PhysRevAccelBeams.25.011301>
18. D'Arcy R., Chappell J., Beinortaitė J., Diederichs S., Boyle G., Foster B., et al. Recovery time of a plasma-wakefield accelerator. *Nature*. 2022;603:58–62. <https://doi.org/10.1038/s41586-021-04348-8>
19. Bagdasarov G.A., Kruchinin K.O., Molodozhentsev A.Yu., Sasorov P.V., Bulanov S.V., Gasilov V.A. Discharge Plasma Formation in Square Capillary with Gas Supply Channels. *Phys. Rev. Res.* 2022;4:013063. <https://doi.org/10.1103/PhysRevResearch.4.013063>
20. Gasilov V.A., Boldarev A.S., Olkhovskaya O.G., Boykov D.S., Sharova Yu.S., Savenko N.O, et al. MARPLE: software for multiphysics modelling in continuous media. *Numerical Methods and Programming*. 2023;24(4):316–338. <https://doi.org/10.26089/NumMet.v24r423>
21. Sasorov P., Bagdasarov G., Bobrova N., Grittani G., Molodozhentsev A., Bulanov S.V. Capillary discharge in the high repetition rate regime. *Physical Review Research*. 2024;6:013290. <https://doi.org/10.1103/PhysRevResearch.6.013290>
22. Savenko N.O. On the difference approximation of gasdynamic flows on the free boundary of the computational domain. *Preprints of the Institute of Applied Mathematics named after M.V. Keldysh*. 2023;51:28. (in Russ.). <https://doi.org/10.20948/prepr-2023-51>
23. Leemans W.P., Nagler B., Gonsalves A.J., Toth Cs., Nakamura K., Geddes C.G.R., et al. GeV electron beams from a centimeter-scale accelerator. *Nature physics*. 2006;2:696–699. <https://doi.org/10.1063/1.2718524>
24. Braginsky S.I., Leontovich M.A. (eds). Phenomena of transport in plasma. In: *Questions of plasma theory*. Issue 1. GosAtomIzdat. 1963. P. 183–272. (in Russ.).

Received 04.06.2024

Revised 24.06.2024

Accepted 25.06.2024

About the Authors:

Vladimir A. Gasilov, Principal Researcher, Keldysh Institute of Applied Mathematics of Russian Academy of Sciences (4, Miusskaya sq., Moscow, 125047, RF), [ORCID](#), [MathNet](#), vgasilov@keldysh.ru

Nikita O. Savenko, Junior Researcher, Keldysh Institute of Applied Mathematics of Russian Academy of Sciences (4, Miusskaya sq., Moscow, 125047, RF), [ORCID](#), [MathNet](#), savenkonkt@gmail.com

Yulia S. Sharova, Junior Researcher, Keldysh Institute of Applied Mathematics of Russian Academy of Sciences (4, Miusskaya sq., Moscow, 125047, RF), [ORCID](#), [MathNet](#), yulia-shar@mail.ru

Claimed contributor-ship:

All authors have made an equivalent contribution to the preparation of the publication.

Conflict of interest statement

The authors do not have any conflict of interest.

All authors have read and approved the final manuscript.

Поступила в редакцию 04.06.2024

Поступила после рецензирования 24.06.2024

Принята к публикации 25.06.2024

Об авторах:

Гасилов Владимир Анатольевич, главный научный сотрудник, Институт прикладной математики им. М.В. Келдыша Российской академии наук (РФ, 125047, Москва, Миусская пл., 4), [ORCID](#), [MathNet](#), vgasilov@keldysh.ru

Савенко Никита Олегович, младший научный сотрудник, Институт прикладной математики им. М.В. Келдыша Российской академии наук (РФ, 125047, Москва, Миусская пл., 4), [ORCID](#), [MathNet](#), savenkonkt@gmail.com

Шарова Юлия Сергеевна, младший научный сотрудник, Институт прикладной математики им. М.В. Келдыша Российской академии наук (РФ, 125047, Москва, Миусская пл., 4), [ORCID](#), [MathNet](#), yulia-shar@mail.ru

Заявленный вклад соавторов:

Все авторы сделали эквивалентный вклад в подготовку публикации.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Все авторы прочитали и одобрили окончательный вариант рукописи.

INFORMATION TECHNOLOGY ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ



UDC 519.6

Original Theoretical Research

<https://doi.org/10.23947/2587-8999-2024-8-2-60-67>


Locating the Interface between Different Media Based on Matrix Ultrasonic Sensor Data Using Convolutional Neural Networks

Alexey V. Vasyukov

Moscow Institute of Physics and Technology (National Research University), Dolgoprudny, Russian Federation

✉ vasyukov.av@mipt.ru

Abstract

Introduction. The study focuses on modelling the process of ultrasound medical examination in a heterogeneous environment with regions of significantly different sound speeds. Such scenarios typically arise when visualizing brain structures through the skull. The aim of this work is to compare possible approaches to determining the interface between acoustically contrasting media using convolutional neural networks.

Materials and Methods. Numerical modelling of the direct problem is performed, obtaining synthetic calculated ultrasonic images based on known geometry and rheology of the area as well as sensor parameters. The calculated images reproduce distortions and artifacts typical for setups involving the skull wall. Convolutional neural networks of 2D and 3D structures following the UNet architecture are used to solve the inverse problem of determining the interface between media based on a sensor signal. The networks are trained on computational datasets and then tested on individual samples not used in training.

Results. Numerical B-scans for characteristic setups were obtained. The possibility of localizing the aberrator boundary with good quality for both 2D and 3D convolutional networks was demonstrated. A higher quality result was obtained for the 3D network in the presence of significant noise and artifacts in the input data. It was established that the 3D architecture network can provide the shape of the interface between media in 0.1 seconds.

Discussion and Conclusions. The results can be used for the development of transcranial ultrasound technologies. Rapid localization of the skull boundary can be incorporated into imaging algorithms to compensate for distortions caused by differences in sound velocities in bone and soft tissues.

Keywords: transcranial ultrasound, matrix probe, aberrations, mathematical modelling, grid-characteristic method, convolutional networks

Funding information. The work was carried out with financial support from the Russian Science Foundation (project 22-11-00142).

For citation. Vasyukov A.V. Locating the interface between different media based on matrix ultrasonic sensor data using convolutional neural networks. *Computational Mathematics and Information Technologies*. 2024;8(2):60–67. <https://doi.org/10.23947/2587-8999-2024-8-2-60-67>

Оригинальное теоретическое исследование

Определение границы раздела сред по трёхмерным данным матричного ультразвукового датчика с использованием свёрточных нейронных сетей

А.В. Васюков

Московский физико-технический институт (национальный исследовательский университет),

г. Долгопрудный, Российская Федерация

✉ vasyukov.av@mipt.ru

Аннотация

Введение. Работа посвящена моделированию процесса ультразвукового медицинского исследования в гетерогенной среде, в которой присутствуют области с существенно разной скоростью звука. Такие постановки задач

возникают, например, при визуализации структур мозга через череп. Целью данной работы является сравнение возможных подходов к определению границы раздела акустически контрастных сред с использованием свёрточных нейронных сетей.

Материалы и методы. В работе выполняется численное моделирование прямой задачи — получение синтетических расчётных ультразвуковых изображений по известной геометрии и реологии области, а также параметрам датчика. На расчётных изображениях воспроизводятся искажения и артефакты, типичные для постановок со стенкой черепа. Для решения обратной задачи определения границы раздела сред по сигналу с датчика используются свёрточные нейронные сети 2D и 3D структуры, следующие общей архитектуре UNet. Сети обучаются на наборах расчётных данных, после чего тестируются на отдельных примерах, не использованных при обучении.

Результаты исследования. Получены расчётные В-сканы для характерных постановок. Показана возможность локализации границы аббератора с хорошим качеством как для 2D, так и для 3D свёрточных сетей. Показано более высокое качество результата для 3D сетей в случае наличия значительного шума и артефактов во входных данных. Установлено, что сеть 3D архитектуры может обеспечить получение формы границы раздела сред за 0,1 секунды.

Обсуждение и заключения. Результаты работы могут быть использованы для развития технологий транскраниального ультразвукового исследования. Быстрая локализация границы стенки черепа может быть включена в алгоритмы построения изображения для компенсации искажений, вызванных различием скоростей звука в костных и в мягких тканях.

Ключевые слова: транскраниальное УЗИ, матричный датчик, абберации, математическое моделирование, сеточно-характеристический метод, свёрточные сети

Финансирование. Работа выполнена при финансовой поддержке Российского научного фонда (код проекта 22-11-00142).

Для цитирования. Васюков А.В. Определение границы раздела сред по данным матричного ультразвукового датчика с использованием свёрточных нейронных сетей. *Computational Mathematics and Information Technologies*. 2024;8(2):60–67. <https://doi.org/10.23947/2587-8999-2024-8-2-60-67>

Introduction. This study addresses the problem of ultrasound image formation in a heterogeneous medium with regions of significantly different sound speeds. This setup is aimed at applications in visualizing brain structures through the skull bones. Despite years of medical technology development, this specific task remains extremely challenging, as existing methods have many limitations and require highly skilled specialists.

The problem arises from the fact that typical algorithms used in commercially available equipment assume that the sound speed in the area of interest changes minimally. This assumption is valid for soft tissues. However, when examining the brain through the skull, this basic assumption fails, leading to highly distorted images using traditional ultrasound approaches [1].

This study focuses on determining the boundary between two media-rigid (model skull wall) and soft (model brain tissue). The proposed solution method must operate in near real-time to ensure practical application. In the future, rapid localization of the skull boundary could be included in imaging algorithms to compensate for distortions caused by differences in sound velocities between bone and soft tissues.

Convolutional neural networks are considered for this task due to their extensive use in related biomedical tasks and their ability to operate at high speeds. Previous studies [2–6] have demonstrated the effectiveness of convolutional networks for ultrasound imaging and elastography. However, using this general approach requires careful calibration for each specific task [7].

Materials and Methods. For the direct problem, numerical modelling of the ultrasound pulse propagation in a sample is performed to obtain synthetic calculated ultrasonic images based on known geometry and rheology of the area and sensor parameters.

The medium is described using the acoustic approximation [8], a significant simplification compared to the full system of elastic equations, including only longitudinal waves. This approach is widely used for describing ultrasound pulses in biological tissues, as the attenuation coefficient for shear waves in ultrasound is four orders of magnitude higher than for longitudinal waves [9].

The numerical solution of the direct problem uses the ray tracing method with wavefront reconstruction [1], allowing the calculation of ultrasound images qualitatively and quantitatively corresponding to experimental data. The method describes reflections from extended boundaries and point reflectors. In this study, the boundary between layers and large pores are modeled as extended boundaries, while small reflectors are considered point sources. After recording the reflected signal, it is processed, and B-scans are constructed using algorithms from [10].

The inverse problem is to determine the shape of the boundary between acoustically contrasting layers based on the sensor's recorded signal. The input data for the inverse problem is the response from the medium registered by the matrix ultrasonic sensor. The output is the position of the boundary between the two media.

Convolutional neural networks are used to solve the inverse problem of determining the boundary based on the sensor signal. A synthetic dataset was generated from 1024 direct problem calculations for network training. Separate examples not included in the training set were used for testing.

This study investigates both 2D and 3D networks to compare results. All convolutional networks follow the UNet architecture [11]. The depth of both 2D and 3D networks is four blocks.

For the 2D network, the three-dimensional data is represented as a set of two-dimensional slices. Each slice is processed with three channels — the target slice and two adjacent slices — providing the network with some three-dimensional context [12, 13].

For the 3D network, three-dimensional data is input using a patch-based approach [14, 15], allowing flexible memory management on the GPU when processing large input data.

Results. The direct problem setup involves calculating the propagation of the ultrasound signal in an area containing a boundary between acoustically contrasting layers. The calculation area is a parallelepiped. The upper face corresponds to the external boundary of the area where the matrix ultrasonic sensor is located. Outside the contact zone with the sensor, the upper face is modeled as a free surface. The other three boundaries are set as non-reflective boundary conditions.

The boundary between the two acoustically contrasting layers is assumed to be smooth and may have an arbitrary shape. Additionally, the upper layer contains many small reflectors, creating background noise in the final ultrasound image, and several large pores whose response intensity is comparable to the boundary reflection.

The sound speed in both layers is constant. The upper layer is more rigid, with a sound speed of 30 km/s. The lower layer is softer, with a sound speed of 15 km/s. The number of small reflectors varied from 100 to 2500, and the number of large pores from 5 to 50.

The matrix sensor has a square shape of 24×24 elements, emitting a signal at 3 MHz. The sampling frequency for signal reception is 45 MHz. The final data dimension is $24 \times 24 \times 1024$, where 24×24 are the physical dimensions of the sensor and 1024 are the time samples recorded during the experiment by each sensor element.

Fig. 1 shows the profile of the medium interface in one of the calculations is presented. Four slices of the complete three-dimensional data are shown — the position of the interface under the rows of sensor elements from the 5th to the 8th. The vertical axis represents the 24 elements of the matrix sensor in the given slice. The horizontal axis represents time samples. The image is cropped to the first 400 samples out of a full set of 1024 samples.

Fig. 2 shows the raw ultrasound image for this calculation is demonstrated. The overall “noise”, visually seen as fluctuations in the intensity of the gray background, is associated with a large number of small reflectors in the medium. The interface between media is visible as an area of intense response with varying amplitude. Individual bright responses from large pores can be seen at depths of 50, 70, 90, 110, 130, and especially 230 (the last two slices in the figure). These bright responses significantly interfere with the automatic image processing, as they even exceed the intensity of the response from the desired boundary.

Figs. 3 and 4 show the results of the 2D convolutional network. Figs. 5 and 6 present the results for the 3D network.

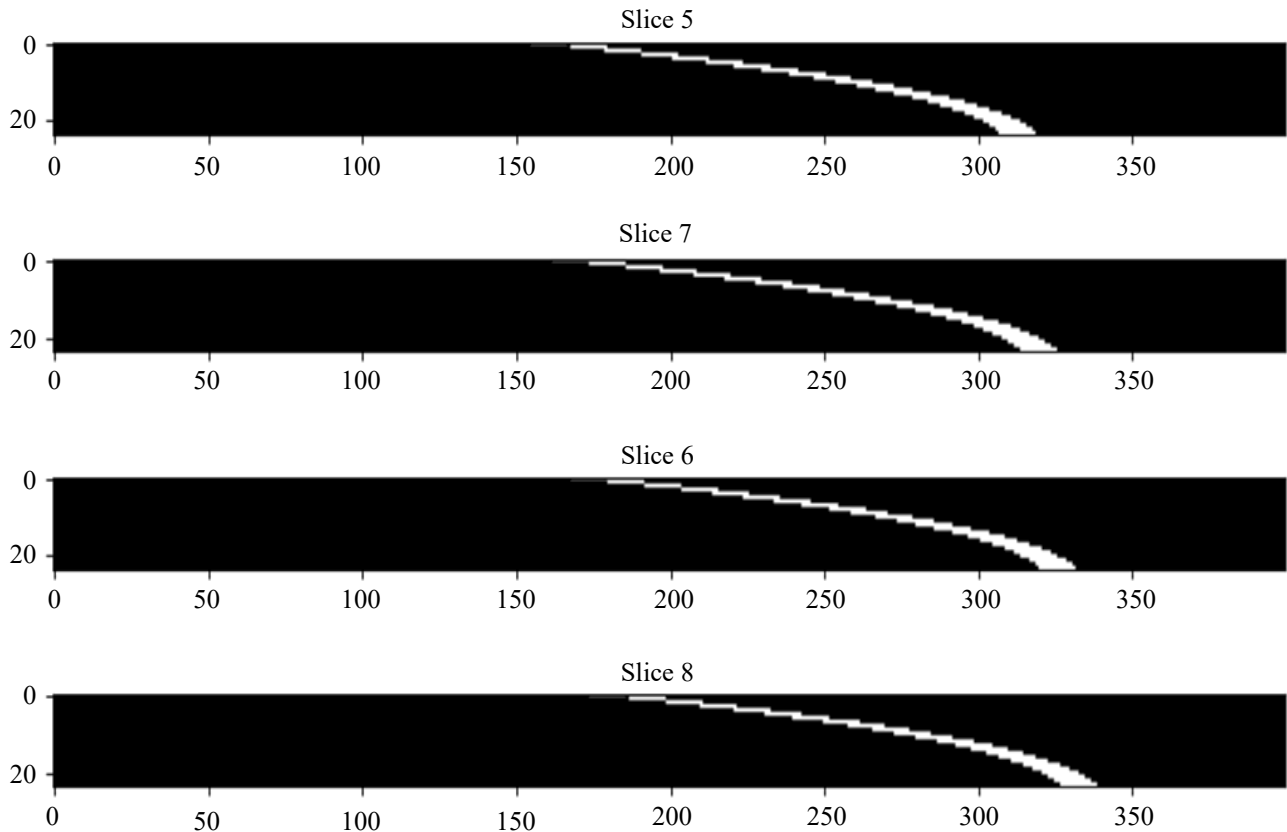


Fig. 1. Location of the medium interface

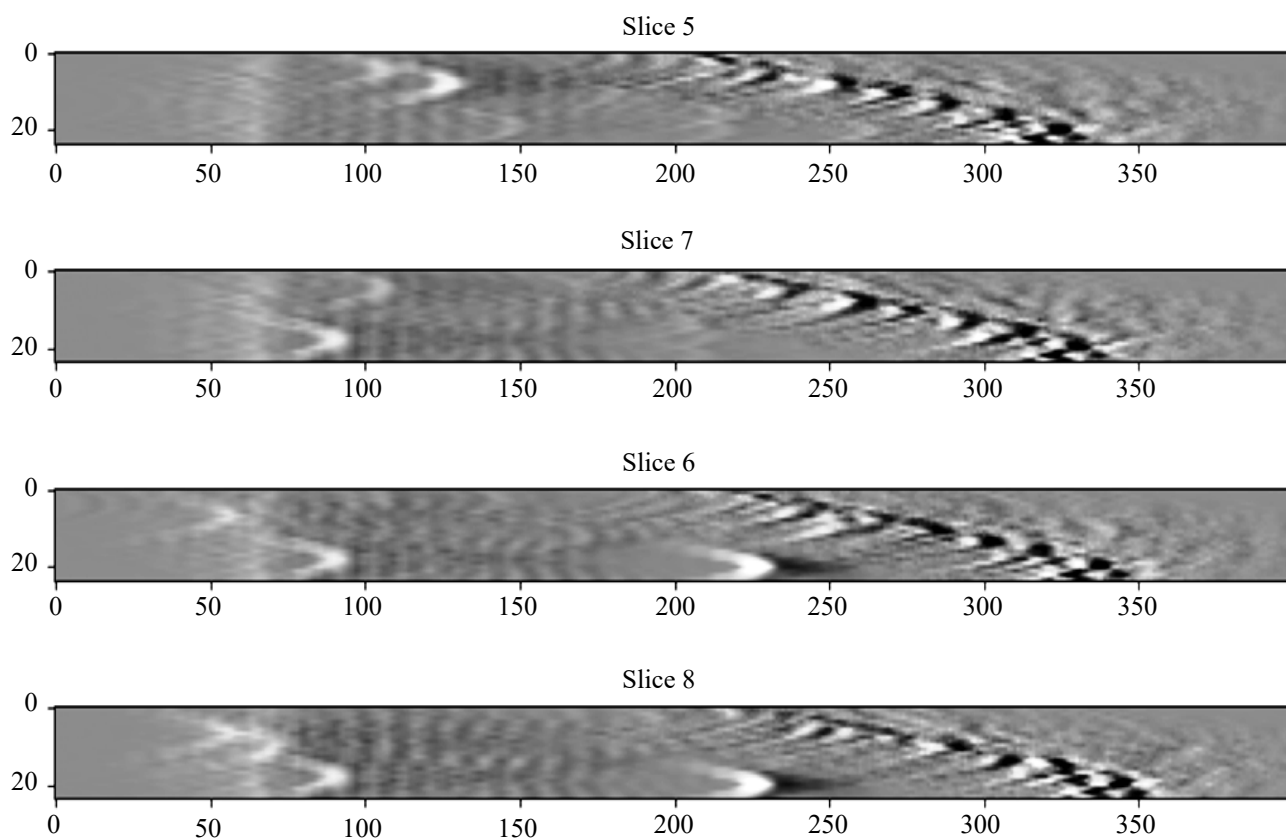


Fig. 2. Ultrasound image (B-scan)

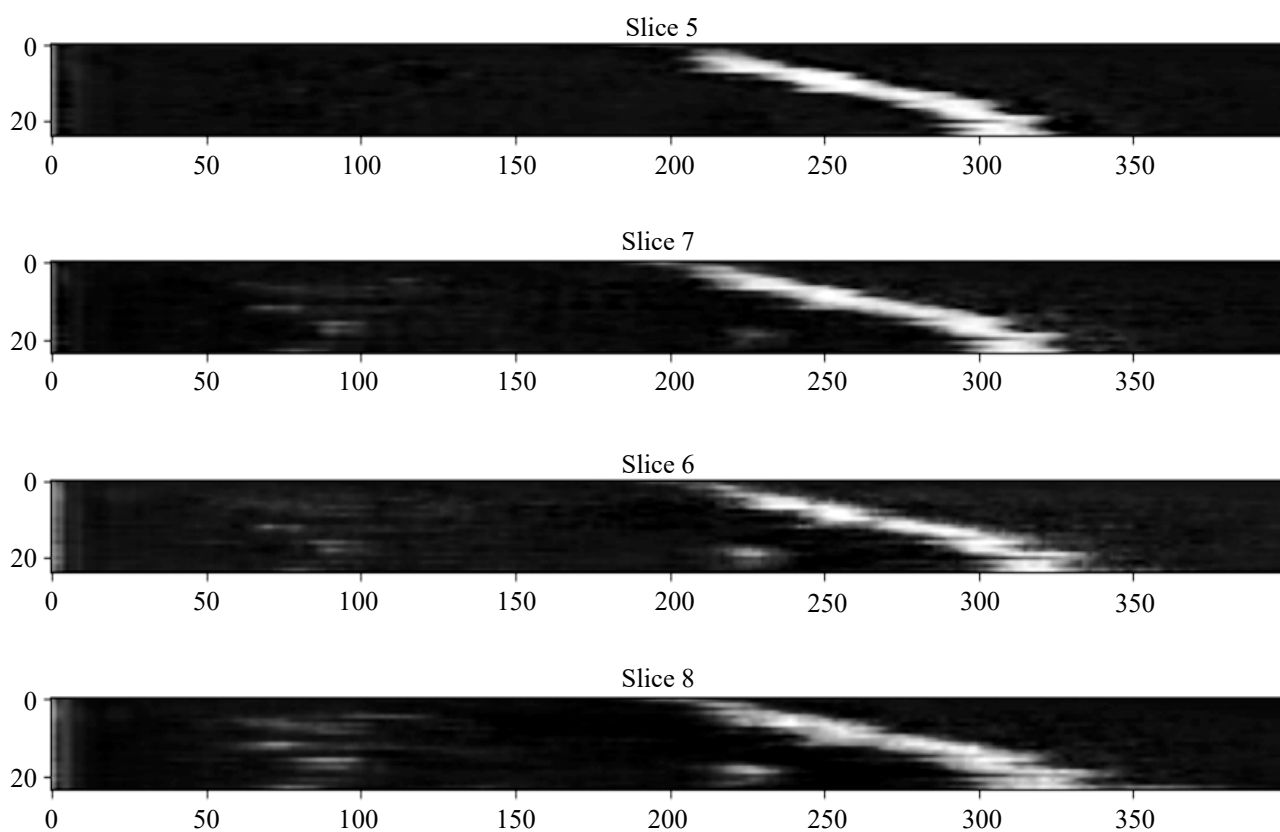


Fig. 3. Predictions of the 2D network in the original form

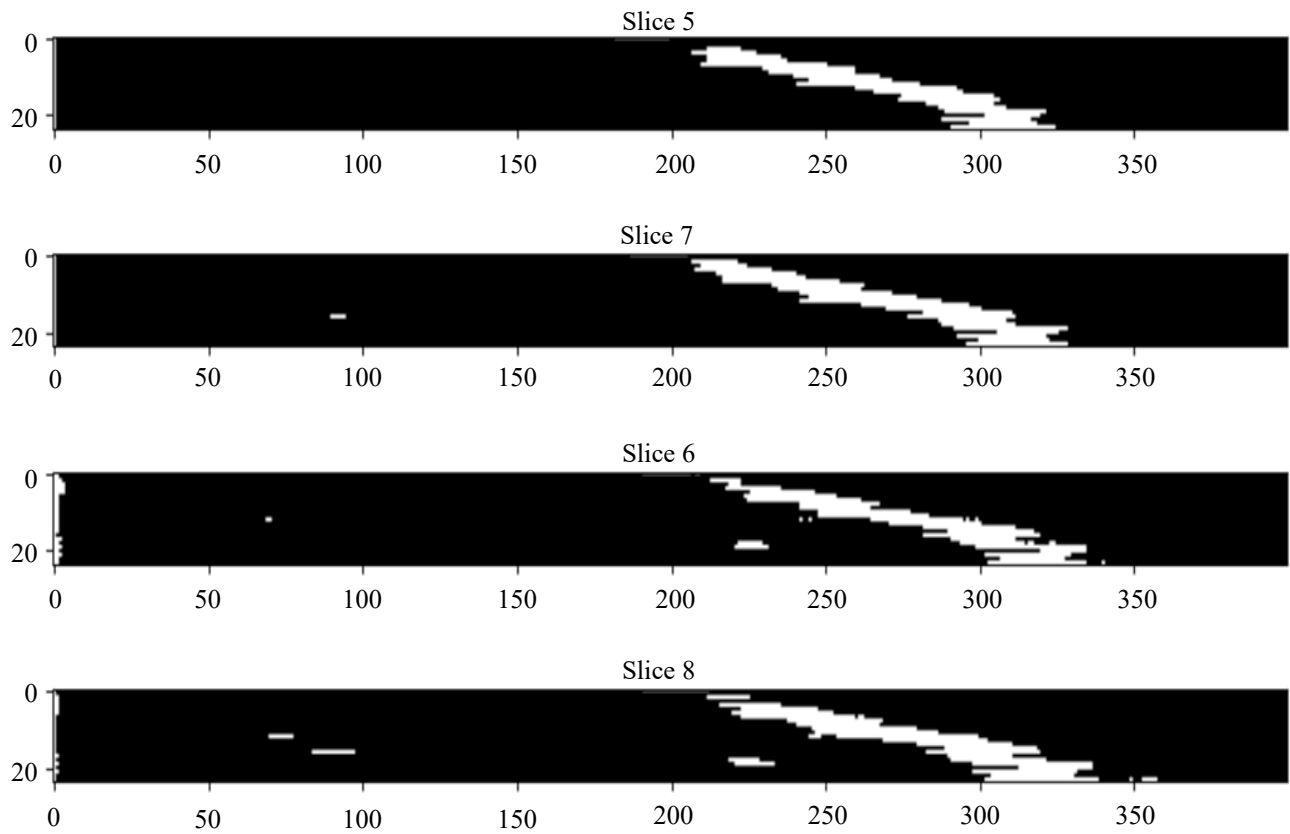


Fig. 4. Predictions of the 2D network after binarization

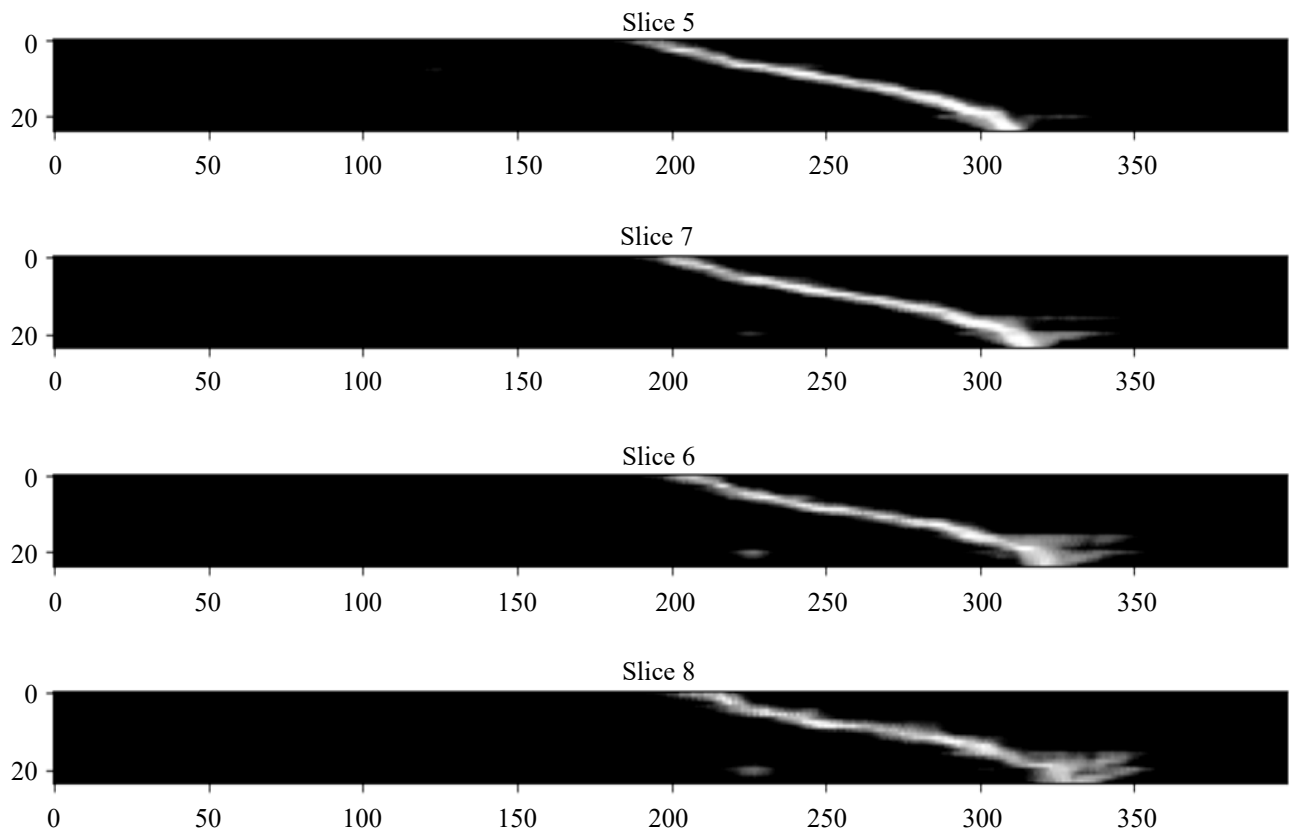


Fig. 5. Predictions of the 3D network in the original form

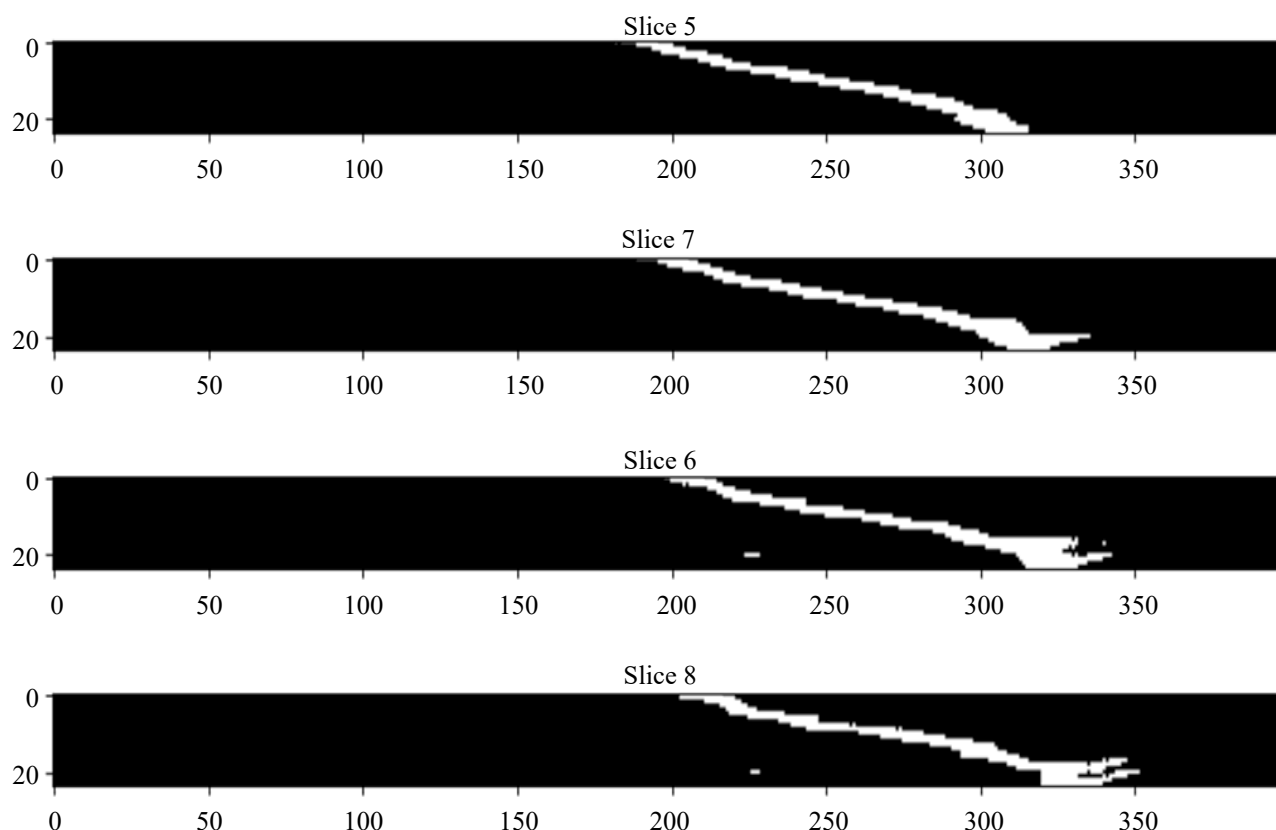


Fig. 6. Predictions of the 3D network after binarization

Discussion and Conclusions. The results show that the 3D convolutional network significantly outperforms the approach of processing three-dimensional data as slices using 2D networks in determining the shape and position of the boundary. Qualitatively, the boundary is generally correctly identified in both scenarios, but the 3D network exhibits substantially less blurring. Notably, the 3D network is almost unaffected by noise and interference in the input signal, both random and those caused by the presence of large bright reflectors. The results of the 2D network (Figs. 3 and 4) show a significant number of detections in the area before the desired boundary — where large pores were located in the object. This is not a random error; the network solves the segmentation task by aiming to detect acoustically contrasting boundaries, and the boundaries of the pores also fall into this category. However, this effect is undesirable. When using the 3D network (Figs. 5 and 6), such problems are virtually eliminated. This is because the three-dimensional structure of the input data allows the convolutional network to fully utilize the spatial information about the reflectors and learn to ignore geometrically small objects.

The total processing time for a single three-dimensional image using the 3D network was about 0.1 seconds on commercially available GPUs. Thus, the possibility of real-time localization of the aberrator boundary with good quality has been demonstrated. This fact can be further used to create new ultrasound imaging algorithms employing methods for compensating distortions caused by differences in sound speeds in tissues.

References

1. Beklemysheva K.A., Grigoriev G.K., Kulberg N.S., Petrov I.B., Vasyukov A.V., Vassilevski Y.V. Numerical simulation of aberrated medical ultrasound signals. *Russian Journal of Numerical Analysis and Mathematical Modelling*. 2018;33(5):277–288. <https://doi.org/10.1515/rnam-2018-0023>
2. Perdios D., Vonlanthen M., Martinez F., Arditi M., Thiran J.P. Single-shot CNN-based ultrasound imaging with sparse linear arrays. In: *2020 IEEE International Ultrasonics Symposium (IUS)*. Las Vegas, NV, USA; 2020. P. 1–4. <https://doi.org/10.1109/IUS46767.2020.9251442>
3. Patel D., Tibrewala R., Vega A., Dong L., Hugenberg N., Oberai A. Circumventing the solution of inverse problems in mechanics through deep learning: Application to elasticity imaging. *Computer Methods in Applied Mechanics and Engineering*. 2019;353:448–466. <https://doi.org/10.1016/j.cma.2019.04.045>

4. Hongya Lu, Haifeng Wang, Qianqian Zhang, Sang Won Yoon, Daehan Won. A 3D Convolutional Neural Network for Volumetric Image Semantic Segmentation. *Procedia Manufacturing*. 2019;39:422–428. <https://doi.org/10.1016/j.promfg.2020.01.386>
5. Potočník B., Šavc M. Deeply-Supervised 3D Convolutional Neural Networks for Automated Ovary and Follicle Detection from Ultrasound Volumes. *Applied Sciences*. 2022;12(3):1246. <https://doi.org/10.3390/app12031246>
6. Brown K., Dormer J., Fei B., Hoyt K. Deep 3D convolutional neural networks for fast super-resolution ultrasound imaging. *Proceedings SPIE 10955, Medical Imaging 2019: Ultrasonic Imaging and Tomography*. 2019;10955:1095502. <https://doi.org/10.1117/12.2511897>
7. Mast T.D., Hinkelman L.M., Metlay L.A., Orr M.J., Waag R.C. Simulation of ultrasonic pulse propagation, distortion, and attenuation in the human chest wall. *J. Acoust. Soc. Amer.* 1999;6:3665–3677. <https://doi.org/10.1121/1.428209>
8. Madsen E.L., Sathoff H.J., Zagzebski J.A. Ultrasonic shear wave properties of soft tissues and tissuelike materials. *J. Acoust. Soc. Am.* 1983;74(5):1346–1355. <https://doi.org/10.1121/1.390158>
9. Vassilevski Y.V., Beklemysheva K.A., Grigoriev G.K., Kulberg N.S., Petrov I.B., Vasyukov A.V. Numerical modelling of medical ultrasound: phantom-based verification. *Russian Journal of Numerical Analysis and Mathematical Modelling*. 2017;32(5):339–346. <https://doi.org/10.1515/rnam-2017-0032>
10. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Lecture Notes in Computer Science*. 2015;9351:234–241. https://doi.org/10.1007/978-3-319-24574-4_28
11. Paserin O., Mulpuri K., Cooper A., Abugharbieh R., Hodgson A. Improving 3D Ultrasound Scan Adequacy Classification Using a Three-Slice Convolutional Neural Network Architecture. In: *CAOS 2018 (EPiC Series in Health Sciences Vol 2)*. Beijing, China; 2018. P. 152–156. <https://doi.org/10.29007/2tct>
12. Jiang M., Spence J.D., Chiu B. Segmentation of 3D ultrasound carotid vessel wall using U-Net and segmentation average network. In: *42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. Montreal, QC, Canada; 2020. P. 2043–2046. <https://doi.org/10.1109/EMBC44109.2020.9175975>
13. Zheng Y., Liu D., Georgescu B., Nguyen H., Comaniciu D. 3D deep learning for efficient and robust landmark detection in volumetric data. *Lecture Notes in Computer Science*. 2015;9349:565–572. https://doi.org/10.1007/978-3-319-24553-9_69
14. Ghimire K., Chen Q., Feng X. Patch-Based 3D UNet for Head and Neck Tumor Segmentation with an Ensemble of Conventional and Dilated Convolutions. *Lecture Notes in Computer Science*. 2021;12603:78–84. https://doi.org/10.1007/978-3-030-67194-5_9
15. Coupeau P., Fasquel J.B., Mazerand E., Menei P., Montero-Menei C.N., Dinomais M. Patch-based 3D U-Net and transfer learning for longitudinal piglet brain segmentation on MRI. *Computer Methods and Programs in Biomedicine*. 2022;214:106563. <https://doi.org/10.1016/j.cmpb.2021.106563>

Received 06.04.2024

Revised 26.04.2024

Accepted 27.04.2024

About the Author:

Alexey V. Vasyukov, Senior Research Fellow at the Department of Informatics and Computational Mathematics, Moscow Institute of Physics and Technology (National Research University) (9, Institutsky Lane, Dolgoprudny, 141701, RF), PhD in Physics and Mathematics, [ORCID](https://orcid.org/0000-0001-9000-0001), vasyukov.av@mipt.ru

Conflict of interest statement

The author does not have any conflict of interest.

The author has read and approved the final manuscript.

Поступила в редакцию 06.04.2024

Поступила после рецензирования 26.04.2024

Принята к публикации 27.04.2024

Об авторе:

Васюков Алексей Викторович, старший научный сотрудник кафедры информатики и вычислительной математики Московского физико-технического института (национального исследовательского университета) (РФ, 141701, Долгопрудный, Институтский переулок, 9), кандидат физико-математических наук, [ORCID](#), vasyukov.av@mipt.ru

Конфликт интересов

Автор заявляет об отсутствии конфликта интересов

Автор прочитал и одобрил окончательный вариант рукописи.

INFORMATION TECHNOLOGY ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ



Original Theoretical Research



UDC 004.032.26

<https://doi.org/10.23947/2587-8999-2024-8-2-68-79>

Application of Neural Networks to Solve the Dirichlet Problem for Areas of Complex Shape

Alexander V. Galaburdin

Don State Technical University, Rostov-on-Don, Russian Federation

✉ Galaburdin@mail.ru

Abstract

Introduction. Many mathematical problems are reduced to solving partial differential equations (PDEs) in domains of complex shapes. Existing analytical and numerical methods do not always provide efficient solutions for such problems. Recently, neural networks have been successfully applied to solve PDEs, typically addressing boundary value problems for domains with simple shapes. This paper attempts to construct a neural network capable of effectively solving boundary value problems for domains of complex shapes.

Materials and Methods. A method for constructing a neural network to solve the Dirichlet problem for regions of complex shape is proposed. Derivatives of singular solutions of the Laplace equation are accepted as activation functions. Singular points of these solutions are distributed along closed curves encompassing the boundary of the domain. The adjustment of the network weights is reduced to minimizing the root-mean-square error during training.

Results. The results of solving Dirichlet problems for various complex-shaped domains are presented. The results are provided in tables, comparing the exact solution and the solution obtained using the neural network. Figures show the domain shapes and the locations of points where the solutions were determined.

Discussion and Conclusion. The presented results indicate a good agreement between the obtained solution and the exact one. It is noted that this method can be easily applied to various boundary value problems. Methods for enhancing the efficiency of such neural networks are suggested.

Keywords: Dirichlet problem, complex-shaped domain, neural networks

For citation. Galaburdin A.V. Application of Neural Networks to Solve the Dirichlet Problem for Areas of Complex Shape. *Computational Mathematics and Information Technologies*. 2024;8(2):68–79.

<https://doi.org/10.23947/2587-8999-2024-8-2-68-79>

Оригинальное теоретическое исследование

Применение нейронных сетей для решения задачи Дирихле для областей сложной формы

А.В. Галабурдин

Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация

✉ Galaburdin@mail.ru

Аннотация

Введение. Многие задачи в математике сводятся к решению дифференциальных уравнений в частных производных для областей сложной формы. Не всегда существующие аналитические и численные методы позволяют эффективно получить решение подобных задач. В последнее время достаточно успешно для решения дифференциальных уравнений в частных производных применяются нейронные сети. При этом обычно рассматриваются краевые задачи для областей, имеющих простую форму. В данной работе предпринимается попытка построить нейронную сеть, способную эффективно решать краевые задачи для областей сложной формы.

Материалы и методы. Предлагается метод построения нейронной сети для решения задачи Дирихле для областей сложной формы. В качестве активационных функций принимаются производные от сингулярных решений уравнения Лапласа. Сингулярные точки этих решений распределены по замкнутым кривым, охватывающих границу области. Настройка весов сети сводится к минимизации среднеквадратической ошибки обучения.

Результаты исследования. Представлены результаты решения задач Дирихле для различных областей сложной формы. Результаты представлены в виде таблиц, содержащих точное решение и решение, полученное при помощи нейронной сети. На рисунках представлен вид областей и расположение точек, в которых определялось решение.

Обсуждение и заключения. Представленные результаты свидетельствуют о хорошем совпадении полученного решения с точным. Отмечается, что данный метод легко применим к различным краевым задачам. Указываются способы повышения эффективности подобных нейронных сетей.

Ключевые слова: задача Дирихле для области сложной формы, нейронные сети

Для цитирования. Галабурдин А.В. Применение нейронных сетей для решения задачи Дирихле для областей сложной формы. *Computational Mathematics and Information Technologies*. 2024;8(2):68–79. <https://doi.org/10.23947/2587-8999-2024-8-2-68-79>

Introduction. Differential equations in partial derivatives are often used in modelling various phenomena. The domains in which these differential equations are defined often have sufficiently complex shapes, making it difficult or impossible to apply known methods effectively. The rapid development of computer technology has allowed for the use of various machine learning methods in solving PDEs.

Recently, the neural network method, whose theoretical foundations were laid in the mid-20th century by A.N. Kolmogorov, has been increasingly used to solve such problems. These methods typically use well-studied differential equations that are relatively simple to solve. Many developers apply boundary value problems for the Laplace equation for this purpose.

For example, the work [2] assesses the quality of approximate solutions to the Laplace equation constructed using neural networks. In [3], a neural network is used to solve the problem of membrane deflection. The article [4] discusses the numerical solution of the Poisson equation in a two-dimensional domain using the Galerkin method and the Ritz method with deep neural networks. Various approaches to training radial-basis neural networks for solving the Poisson equation are discussed in [5].

The study [6] proposes a network architecture that allows solving Laplace, Poisson, heat conduction, and wave equations for rectangular domains. Methods for solving PDEs using radial-basis neural networks, feedforward networks, and modified neural networks are considered in [7]. Using a perceptron-type neural network with a single hidden layer, [8] obtains an analytical approximation of solutions for parabolic-type PDEs.

The use of radial-basis functions in implementing the finite element method with neural networks is explored in [9]. Studies [10, 11] vary the parameters of radial-basis functions when training radial-basis neural networks.

The method of physics-informed neural networks is currently gaining popularity for solving PDEs [12]. The study [13] describes algorithms for using physics-informed neural networks to solve classical mechanics problems.

Artificial neural networks were used to solve the Navier-Stokes equations in [14]. The article [15] investigates approaches to solving heat and mass transfer problems based on a perceptron-type neural network.

The examples above illustrate a wide range of problems solved using neural networks and the various approaches to applying neural networks to solve different boundary value problems. Neural networks are more commonly applied to solving boundary value problems for domains of simple shapes. This study aims to propose an approach for using neural networks to solve boundary value problems for complex-shaped domains.

Materials and Methods. Consider the Dirichlet problem for a plane region G , bounded by a smooth closed curve γ . One effective method for solving this problem is the boundary integral equation method. To obtain the corresponding boundary integral equation, Green's formula can be used:

$$u = \frac{1}{2\pi} \int_{\gamma} \frac{\partial u}{\partial n} U \, d\gamma - \frac{1}{2\pi} \int_{\gamma} \frac{\partial u}{\partial n} u \, d\gamma.$$

Here, U is the singular solution of the Laplace equation.

Using a quadrature formula for calculating integrals, this relationship can be represented as:

$$u_i = \frac{1}{2\pi} \sum_{k=1}^N C_k \left[\frac{\partial u}{\partial n} \right]_k [U]_{ik} - \frac{1}{2\pi} \sum_{k=1}^N C_k [u]_k \left[\frac{\partial U}{\partial n} \right]_{ik}, \quad (1)$$

where u_i is the value of u at the i -th point of the boundary γ , C_k are the coefficients of the quadrature formula.

In this expression, $[U]_{ik}$ and $\left[\frac{\partial U}{\partial n}\right]_{ik}$ can be considered as activation functions, while $C_k\left[\frac{\partial u}{\partial n}\right]_k$ and $C_k[u]_k$ can be considered as weights.

By requiring the fulfillment of the relationship in each point of the boundary for all functions of the training set, a system of equations for determining the weights can be obtained using the least squares method. However, these systems of equations are ill-conditioned. To improve the conditioning of these systems, the singularity of $[U]_{ik}$ and $\left[\frac{\partial U}{\partial n}\right]_{ik}$, can be increased by shifting the integration contour some distance away from the boundary γ .

The Dirichlet problem solution can then be sought in the form:

$$u(x) = \sum_{k=1}^N w_k f(s_k) U(x, \sigma_k) + \sum_{k=1}^N v_k f(s_k) V(x, \tau_k),$$

where $f(s_k)$ is the value of the unknown function u on the boundary; $U(x, \sigma_k)$ и $V(x, \tau_k)$ are activation functions; σ_k and τ_k are points on closed curves γ_1 and γ_2 , encompassing the boundary γ ; x is a point in the domain G .

The curves γ_1 and γ_2 are similar to the contour γ and are obtained by shifting each point in the direction of the outward normal to the boundary by distances ε_1 and ε_2 respectively.

During network training, weights are adjusted and the values ε_1 and ε_2 , are determined by minimizing the error functional:

$$J(w_k, v_k, \varepsilon_1, \varepsilon_2) = \sum_{j=1}^M \sum_{i=1}^N \left\{ \sum_{k=1}^N w_k f_k^j U(x_i, \sigma_k) + v_k f_k^j V(x_i, \tau_k) - f_i^j \right\}^2,$$

where x_i is the coordinate of the i -th point of the boundary contour γ ; f_i^j is the boundary value of the j -th function in the training set at point x_i .

From the relations $\frac{\partial J}{\partial w_m} = 0$ и $\frac{\partial J}{\partial v_m} = 0$, $m = 1, 2, \dots, N$ a system of linear equations for determining w_m and v_m can be obtained.

The values ε_1 and ε_2 are determined by simple iteration. Assuming $\varepsilon_2 = \varepsilon_1 + 1$, the values of $\varepsilon_1 = a + hj$, $j = 1, 2, \dots, L$ are chosen. The value of ε_1 , that provides the best result is selected. After that, all neural network parameters are determined and its configuration is completed.

The accuracy of the obtained solution can be assessed by comparing the values of u on the boundary calculated using the neural network:

$$\tilde{u}(s_i) = \sum_{k=1}^N w_k f(s_k) U(s_i, \sigma_k) + \sum_{k=1}^N v_k f(s_k) V(s_i, \tau_k)$$

with the given boundary conditions $f(s)$.

The defined network parameters do not always ensure the desired accuracy of the neural network solution. In this case, the required accuracy can be achieved by iterative refinement of the obtained result:

$$\Delta u^o(s_i) = f(s_i), \quad u_t^o(s_i) = f(s_i),$$

$$\Delta v^{n+1}(s_i) = \sum_{k=1}^N w_k \Delta u^n(s_k) U(s_i, \sigma_k) + \sum_{k=1}^N v_k \Delta u^n(s_k) V(s_i, \tau_k),$$

$$\Delta u^{n+1}(s_i) = \Delta u^{n+1}(s_i) - \Delta v^{n+1}(s_i), \quad u_t^{n+1}(s_i) = u_t^{n+1}(s_i) + \Delta u^{n+1}(s_i),$$

$u_t^{n+1}(s_i)$ are the values of the refined solution at the boundary of the region.

The refinement process continues until the specified accuracy is achieved, i. e. $\frac{\|\Delta u^{n+1}(s_i)\|}{\|u_t^{n+1}(s_i)\|} < \delta$ or the value $\frac{\|\Delta u^{n+1}(s_i)\|}{\|u_t^{n+1}(s_i)\|}$.

starts to increase.

After this, the value of the solution at any point x in the domain G can be computed using the formula:

$$\tilde{u}(x) = \sum_{k=1}^N w_k u_k u_t(s_k) U(x, \sigma_k) + \sum_{k=1}^N v_k u_t(s_k) V(x, \tau_k),$$

where $u_t(s_k)$ are the refined values of the unknown function on the boundary γ .

The training set used functions that are solutions to the Laplace equation:

$$r^k \cos \left(\arccos \left(\frac{x}{r} \right) \right) + r^k \sin \left(\arccos \left(\frac{x}{r} \right) \right), r = \sqrt{x^2 + y^2},$$

where $k = 0, 1, 2, 3, \dots, M$.

These functions were specified in different coordinate systems, each rotated relative to the others by an angle that is a multiple of $2\pi/5$.

Results. The presented method was utilized to solve the Dirichlet problem for regions whose boundary γ was defined as:

$$\begin{cases} x = a \cos(t) + g \cos(\alpha t), \\ y = b \sin(t) + q \sin(\alpha t), \end{cases} t \in [0, 2\pi],$$

where a, b, g, q, α are variable parameters.

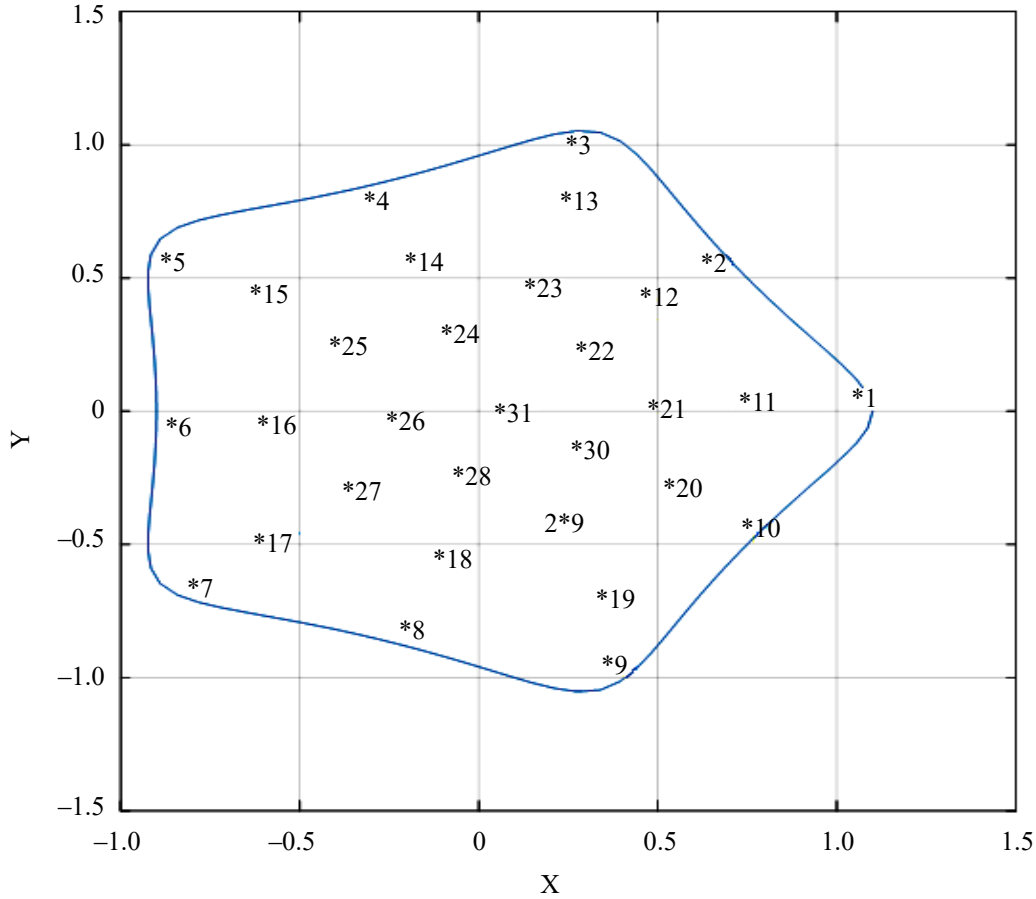


Fig. 1. Region G1

Figure 1 shows the region whose boundary corresponds to $a = 1, b = 1, g = 0.1, q = -0.1, \alpha = 4$. The numbered stars indicate the locations of points in region G1 where the exact Dirichlet problem solutions and the values obtained using the neural network with $\varepsilon_1 = 5$ are calculated.

Table 1 presents the calculation results corresponding to the solution

$$u = e^{2.45x} \cos 2.45y. \quad (2)$$

The table includes the point numbers in region G1, their coordinates, the exact solution of the Dirichlet problem, and the solution obtained by the neural network.

Table 2 presents the calculation results corresponding to the solution

$$u = \frac{x^3 + xy^2 + x^2 - y^2 + 5x + 5}{(x+1)^2 + y^2}. \quad (3)$$

in region G1.

Figure 2 shows the region corresponding to $a = 1, b = 1, g = 0.1, q = 0.1, \alpha = 5$. Tables 3 and 4 present the calculation results corresponding to solutions (2) and (3) in region G2 for $\varepsilon_1 = 6.45$. Figure 3 shows region G3, corresponding to $a = 1, b = 1, g = 0.2, q = -0.2, \alpha = 2$. The calculation results corresponding to solutions (2) and (3) in region G3 for (3) $\varepsilon_1 = 6.3$, are presented in Tables 5 and 6.

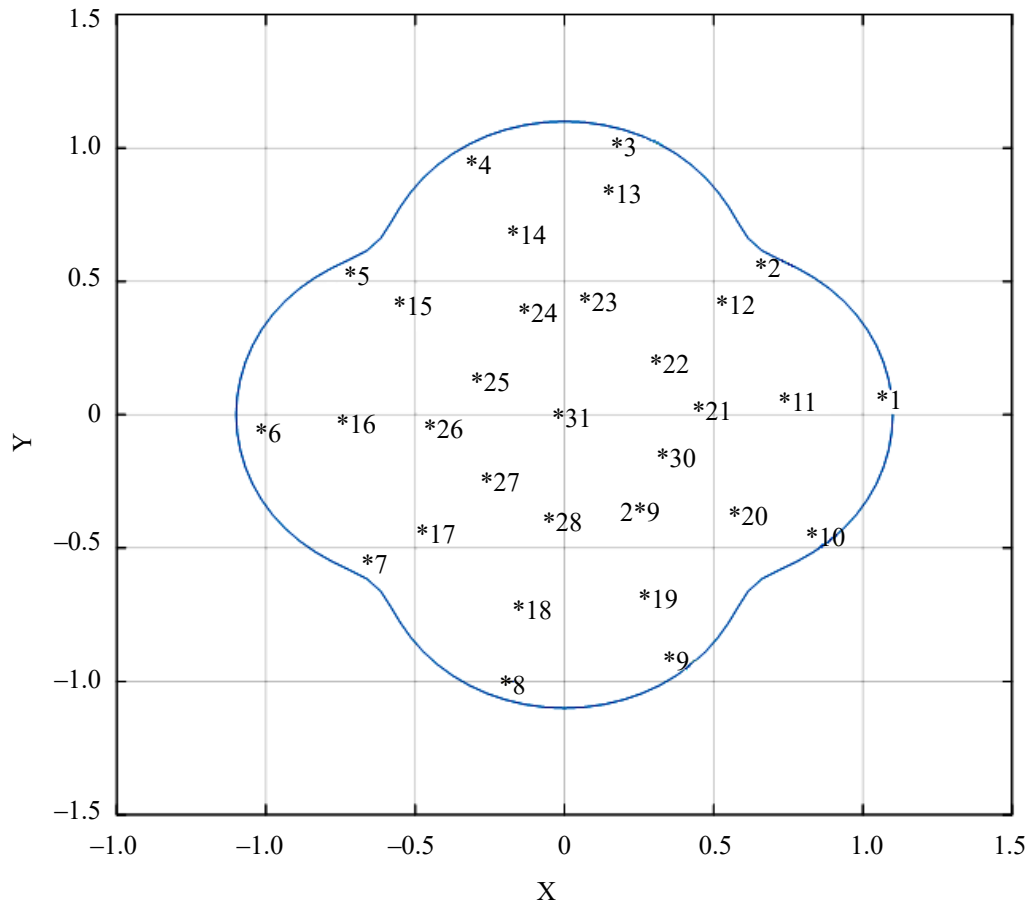


Fig. 2. Region G2

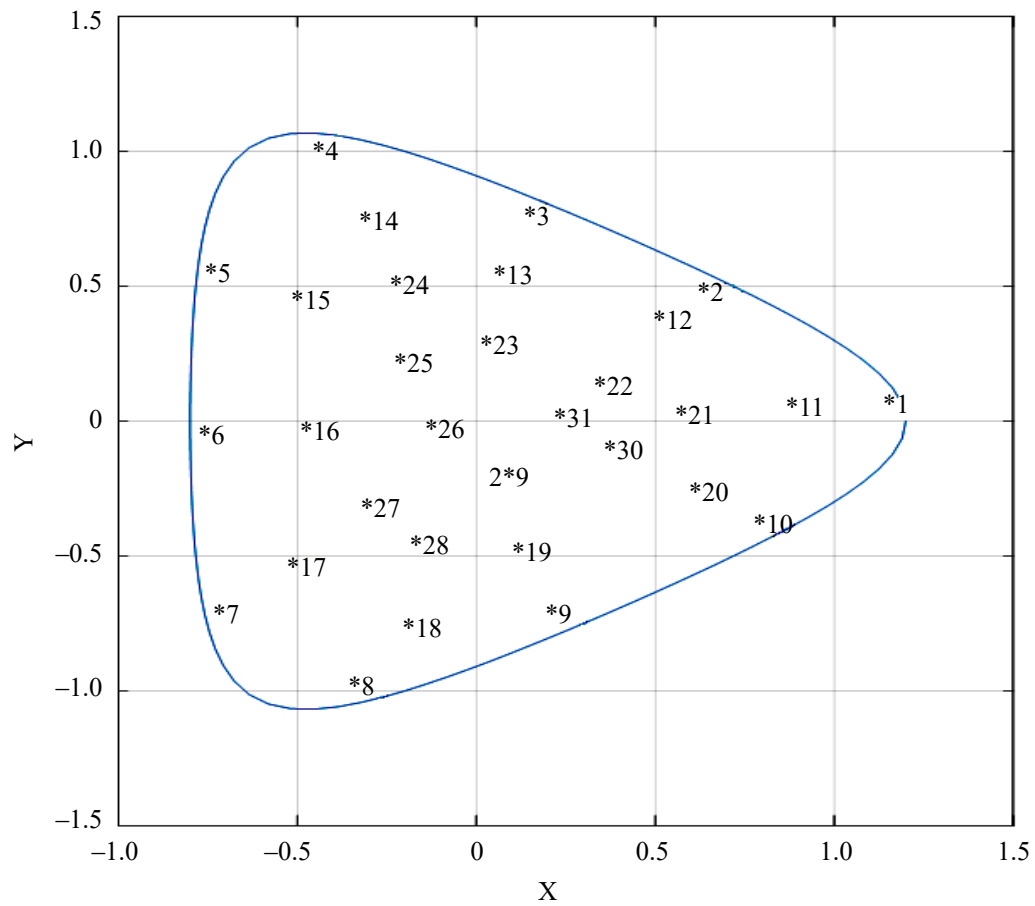


Fig. 3. Region G3

Table 1

The calculation results

Point number	1	2	3	4	5	6	7
x	1.0351	0.6510	0.2626	-0.3216	-0.8728	-0.8497	-0.8020
y	0.0602	0.5496	1.0030	0.7890	0.5597	-0.0620	-0.6571
The exact solution	12.4909	1.0954	-1.4745	-0.1611	0.0234	0.1233	-0.0055
The HC solution	12.477	1.1020	-1.4730	-0.1630	0.0240	0.1240	-0.0090
Point number	8	9	10	11	12	13	14
x	-0.2036	0.3771	0.7239	0.7804	0.4546	0.1996	-0.2272
y	-0.8273	-0.9658	-0.4493	0.0437	0.3866	0.7557	0.5518
The exact solution	0.2675	-1.7992	2.6689	6.7272	1.7790	-0.4516	0.1244
The HC solution	0.2736	-1.7670	2.6512	6.7282	1.7803	-0.4501	0.1245
Point number	15	16	17	18	19	20	21
x	-0.6570	-0.5950	-0.6056	-0.1406	0.2827	0.5081	0.4747
y	0.4233	-0.0455	-0.4941	-0.5800	-0.7287	-0.3129	0.0239
The exact solution	0.1017	0.2313	0.0800	0.1058	-0.4254	2.5011	3.1943
The HC solution	0.1023	0.2312	0.0817	0.1020	-0.4318	2.4936	3.1933
Point number	22	23	24	25	26	27	28
x	0.2190	0.1240	-0.1139	-0.3981	-0.2894	-0.3700	-0.0650
y	0.1909	0.4589	0.2673	0.2597	-0.0257	-0.2984	-0.2831
The exact solution	1.5265	0.5851	0.6001	0.3033	0.4912	0.3007	0.6557
The HC solution	1.5259	0.5857	0.5997	0.3032	0.4903	0.3003	0.6527
Point number	29	30	31				
x	0.1694	0.2492	0.0744				
y	-0.4441	-0.1493	0.0000				
The exact solution	0.7030	1.7196	1.1998				
The HC solution	0.6961	1.7160	1.1980				

Table 2

The calculation results

Point number	1	2	3	4	5	6	7
x	1.0351	0.6510	0.2626	-0.3216	-0.8728	-0.8497	-0.8020
y	0.0602	0.5496	1.0030	0.7890	0.5597	-0.0620	-0.6571
The exact solution	3.9985	3.8121	3.1740	4.1404	7.0650	9.9034	5.9459
The HC solution	4.0010	3.8210	3.1740	4.1330	7.0860	9.8590	5.9370
Point number	8	9	10	11	12	13	14
x	-0.2036	0.3771	0.7239	0.7804	0.4546	0.1996	-0.2272
y	-0.8273	-0.9658	-0.4493	0.0437	0.3866	0.7557	0.5518
The exact solution	3.8595	3.2536	3.8810	4.0331	4.0649	3.7186	4.8021
The HC solution	3.8675	3.2293	3.8793	4.0297	4.0655	3.7168	4.7974
Point number	15	16	17	18	19	20	21
x	-0.6570	-0.5950	-0.6056	-0.1406	0.2827	0.5081	0.4747
y	0.4233	-0.0455	-0.4941	-0.5800	-0.7287	-0.3129	0.0239
The exact solution	6.8777	7.6080	6.2579	4.5410	3.7216	4.0829	4.2223
The HC solution	6.8790	7.5937	6.2519	4.5407	3.7254	4.0835	4.2207
Point number	22	23	24	25	26	27	28

Continuation of table 2

x	0.2190	0.1240	-0.1139	-0.3981	-0.2894	-0.3700	-0.0650
y	0.1909	0.4589	0.2673	0.2597	-0.0257	-0.2984	-0.2831
The exact solution	4.4902	4.3447	5.1277	6.1527	5.9413	5.9450	4.9773
The HC solution	4.4883	4.3425	5.1237	6.1467	5.9351	5.9387	4.9750
Point number	29	30	31				
x	0.1694	0.2492	0.0744				
y	-0.4441	-0.1493	0.0000				
The exact solution	4.3023	4.4661	4.8261				
The HC solution	4.3041	4.4653	4.8236				

Table 3

The calculation results

Point number	1	2	3	4	5	6	7
x	1.0403	0.6832	0.2463	-0.3558	-0.7924	-1.0403	-0.6832
y	0.0633	0.5522	0.9914	0.9278	0.4881	-0.0633	-0.5522
The exact solution	12.6374	1.1533	-1.3835	-0.2702	0.0526	0.0772	0.0406
The HC solution	12.5300	1.1510	-1.3230	-0.2480	0.0360	0.1000	0.0290
Point number	8	9	10	11	12	13	14
x	-0.2463	0.3558	0.7924	0.7591	0.4985	0.1798	-0.2596
y	-0.9914	-0.9278	-0.4881	0.0462	0.4029	0.7235	0.6771
The exact solution	-0.4138	-1.5445	2.5519	6.3819	1.8691	-0.3111	-0.0465
The HC solution	-0.4635	-1.4973	2.5558	6.3787	1.8724	-0.3007	-0.0338
Point number	15	16	17	18	19	20	21
x	-0.5782	-0.7591	-0.4985	-0.1798	0.2596	0.5782	0.4217
y	0.3562	-0.0462	-0.4029	-0.7235	-0.6771	-0.3562	0.0257
The exact solution	0.1559	0.1547	0.1625	-0.1290	-0.1660	2.6505	2.8047
The HC solution	0.1554	0.1509	0.1558	-0.1404	-0.1775	2.6392	2.8042
Point number	22	23	24	25	26	27	28
x	0.2770	0.0999	-0.1442	-0.3212	-0.4217	-0.2770	-0.0999
y	0.2238	0.4019	0.3761	0.1979	-0.0257	-0.2238	-0.4019
The exact solution	1.6820	0.7064	0.4246	0.4027	0.3551	0.4329	0.4331
The HC solution	1.6832	0.7097	0.4288	0.4031	0.3514	0.4279	0.4261
Point number	29	30	31				
x	0.1442	0.3212	-0.0282				
y	-0.3761	-0.1979	0.0000				
The exact solution	0.8608	1.9437	0.9332				
The HC solution	0.8512	1.9369	0.9308				

Fig. 4 and Fig. 5 show graphically obtained results of solving the Dirichlet problem in G_3 for solution (3).

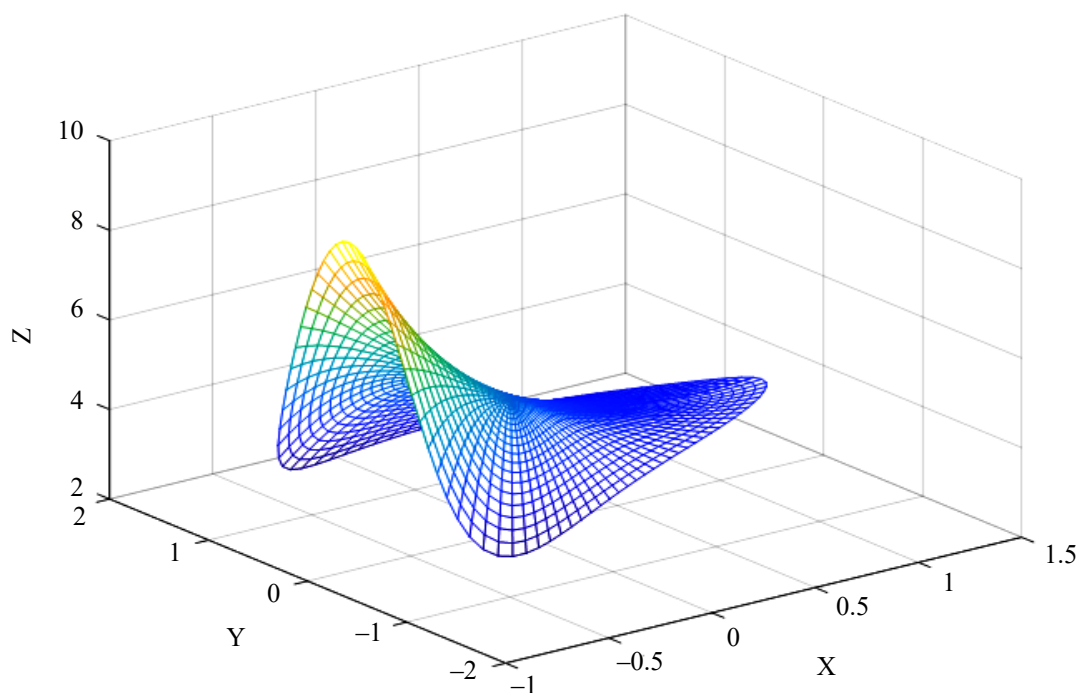
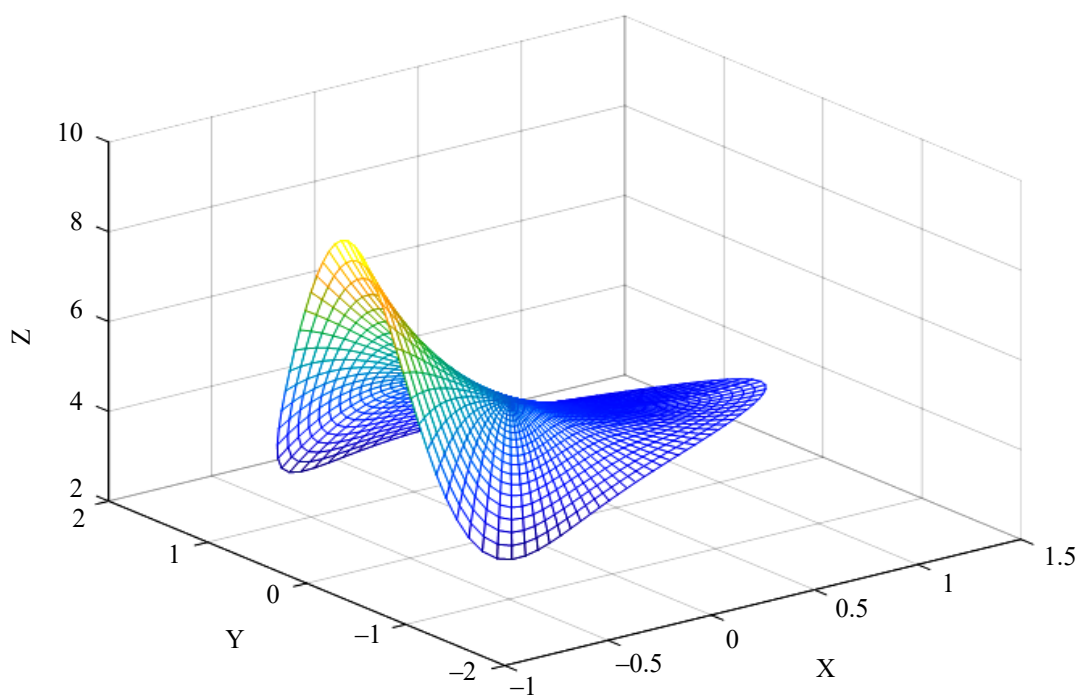
Fig. 4. HC solution in $G3$ corresponding to (3)Fig. 5. The exact solution in $G3$ corresponding to (3)

Table 4

The calculation results

Point number	1	2	3	4	5	6	7
x	1.0403	0.6832	0.2463	-0.3558	-0.7924	-1.0403	-0.6832
y	0.0633	0.5522	0.9914	0.9278	0.4881	-0.0633	-0.5522
The exact solution	3.9985	3.8023	3.1994	3.5918	7.2276	12.5939	6.2589
The HC solution	3.9990	3.7940	3.1690	3.6220	7.2120	12.488	6.2450

Continuation of table 4

Point number	8	9	10	11	12	13	14
x	−0.2463	0.3558	0.7924	0.7591	0.4985	0.1798	−0.2596
y	−0.9914	−0.9278	−0.4881	0.0462	0.4029	0.7235	0.6771
The exact solution	3.2856	3.3263	3.8398	4.0402	4.0187	3.7979	4.4638
The HC solution	3.3086	3.3405	3.8291	4.0326	4.0107	3.7853	4.4515
Point number	15	16	17	18	19	20	21
x	−0.5782	−0.7591	−0.4985	−0.1798	0.2596	0.5782	0.4217
y	0.3562	−0.0462	−0.4029	−0.7235	−0.6771	−0.3562	0.0257
The exact solution	6.7562	8.9677	6.1858	4.1803	3.8289	4.0072	4.2753
The HC solution	6.7422	8.9594	6.1844	4.1866	3.8342	4.0038	4.2690
Point number	22	23	24	25	26	27	28
x	0.2770	0.0999	−0.1442	−0.3212	−0.4217	−0.2770	−0.0999
y	0.2238	0.4019	0.3761	0.1979	−0.0257	−0.2238	−0.4019
The exact solution	4.3835	4.4688	5.0283	5.9328	6.5585	5.7193	4.8676
The HC solution	4.3760	4.4594	5.0170	5.9234	6.5537	5.7168	4.8675
Point number	29	30	31				
x	0.1442	0.3212	−0.0282				
y	−0.3761	−0.1979	0.0000				
The exact solution	4.4325	4.3410	5.0723				
The HC solution	4.4321	4.3377	5.0668				

Table 5

The calculation results

Point number	1	2	3	4	5	6	7
x	1.1387	0.6789	0.1404	−0.4339	−0.7355	−0.7488	−0.7213
y	0.0610	0.4697	0.7826	1.0207	0.5615	−0.0620	−0.6858
The exact solution	16.0958	2.1517	−0.4790	−0.2769	0.0320	0.1579	−0.0187
The HC solution	16.055	2.1310	−0.4540	−0.2700	0.0420	0.1780	−0.0090
Point number	8	9	10	11	12	13	14
x	−0.3119	0.2489	0.7847	0.8836	0.4897	0.0792	−0.3416
y	−0.9953	−0.7209	−0.4028	0.0444	0.3136	0.5423	0.7854
The exact solution	−0.3552	−0.3575	3.7699	8.6620	2.3870	0.2911	−0.1498
The HC solution	−0.3695	−0.3003	3.7043	8.6771	2.3820	0.3008	−0.1382
Point number	15	16	17	18	19	20	21
x	−0.5300	−0.4937	−0.5321	−0.2508	0.1566	0.5793	0.5776
y	0.4312	−0.0454	−0.5297	−0.7550	−0.4856	−0.2725	0.0244
The exact solution	0.1343	0.2965	0.0732	−0.1490	0.5459	3.2463	4.109099
The HC solution	0.1459	0.3115	0.0881	−0.1402	0.5636	3.2274	4.1094
Point number	22	23	24	25	26	27	28
x	0.2626	0.0059	−0.2309	−0.2835	−0.1876	−0.3050	−0.1775
y	0.1262	0.2540	0.5029	0.2748	−0.0254	−0.3423	−0.4667
The exact solution	1.8128	0.8243	0.1887	0.3903	0.6302	0.3166	0.2683
The HC solution	1.8150	0.8318	0.2004	0.4015	0.6414	0.3311	0.2834
Point number	29	30	31				
x	0.0459	0.3328	0.1744				
y	−0.2032	−0.1161	0.0000				

Continuation of table 5

The exact solution	0.9832	2.1689	1.5329				
The HC solution	0.9924	2.1689	1.5373				

Table 6

The calculation results

Point number	1	2	3	4	5	6	7
x	1.1387	0.6789	0.1404	-0.4339	-0.7355	-0.7488	-0.7213
y	0.0610	0.4697	0.7826	1.0207	0.5615	-0.0620	-0.6858
The exact solution	4.0101	3.8766	3.6988	3.1990	6.4195	8.8530	5.4894
The HC solution	4.0110	3.8840	3.6990	3.2160	6.4370	8.8080	5.4920
Point number	8	9	10	11	12	13	14
x	-0.3119	0.2489	0.7847	0.8836	0.4897	0.0792	-0.3416
y	-0.9953	-0.7209	-0.4028	0.0444	0.3136	0.5423	0.7854
The exact solution	3.2886	3.7558	3.9008	4.0079	4.0963	4.2640	4.1824
The HC solution	3.2830	3.7587	3.8986	4.0062	4.0998	4.2648	4.1807
Point number	15	16	17	18	19	20	21
x	-0.5300	-0.4937	-0.5321	-0.2508	0.1566	0.5793	0.5776
y	0.4312	-0.0454	-0.5297	-0.7550	-0.4856	-0.2725	0.0244
The exact solution	6.2152	6.9516	5.7855	4.1727	4.2569	4.0602	4.1378
The HC solution	6.2154	6.9378	5.7897	4.1682	4.2566	4.0600	4.1380
Point number	22	23	24	25	26	27	28
x	0.2626	0.0059	-0.2309	-0.2835	-0.1876	-0.3050	-0.1775
y	0.1262	0.2540	0.5029	0.2748	-0.0254	-0.3423	-0.4667
The exact solution	4.4559	4.8437	4.9521	5.6575	5.5545	5.5934	4.9136
The HC solution	4.4563	4.8424	4.9498	5.6532	5.5499	5.5904	4.9112
Point number	29	30	31				
x	0.0459	0.3328	0.1744				
y	-0.2032	-0.1161	0.0000				
The exact solution	4.8040	4.3626	4.6270				
The HC solution	4.8022	4.3624	4.6263				

In all cases, when clarifying the decision, $M = 75$, $\delta = 0.00025$ were taken and the Euclidean norm was used. The following activation functions were taken

$$U(x, y, t, s) = \frac{\partial^6}{\partial^3 t \partial^3 s} Y, \quad V(x, y, t, s) = \frac{\partial^5}{\partial^3 t \partial^2 s} Y - \frac{\partial^5}{\partial^2 t \partial^3 s} Y,$$

$$Y = \ln \frac{1}{R}, \quad R = \sqrt{(x-t)^2 + (y-s)^2}.$$

Discussion and Conclusion. The presented results convincingly demonstrate that the proposed method for constructing a neural network to solve the Dirichlet problem for regions of complex shapes is highly effective. This method can also be utilized for solving other partial differential equations. It can be easily adapted for solving three-dimensional problems and boundary value problems for multiply connected regions. Its efficiency can be further enhanced by appropriately selecting activation functions (by choosing parameters ε_1 and ε_2), by optimizing the training set selection, and by fine-tuning the weights. All of the above indicates the considerable potential of the proposed method.

References

1. Kolmogorov A.N. On the Representation of Continuous Functions of Several Variables by Superpositions of Continuous Functions of One Variable and Addition. *Doklady Akademii Nauk SSSR*. 1957; 114(5): 953–956. (in Russ.).
2. Varshavchik E.A., Galyautdinova A.R., Sedova Y. S., Tarkhov D.A. Solving Partial Differential Equations for Regions with Constant Boundaries. Artificial Intelligence in Solving Current Social and Economic Problems of the 21st Century. In: *Proceedings of the Third All-Russian Scientific-Practical Conference*. Perm: Publishing House of Perm State National Research University; 2018. pp. 294–303. (in Russ.).
3. Bortkovskaya M.R., Kaverzneva T.T., Semenova D.A., Shishkina I.A., Tarkhov D.A., Udalov P.P. Construction of a Mathematical Model of Membrane Deflection Using the Two-Layer Euler Method Based on a Differential Equation and Experimental Data. Artificial Intelligence in Solving Current Social and Economic Problems of the 21st Century. In: *Proceedings of the Third All-Russian Scientific-Practical Conference*. Perm: Publishing House of Perm State National Research University; 2018. pp. 194–201. (in Russ.).
4. Epifanov A.A. Application of Deep Learning Methods for Solving Partial Differential Equations. *Advances in Cybernetics*. 2020;1(4): 22–28. (In Russ.). <https://doi.org/10.51790/2712-9942-2020-1-4-3>
5. Gorbachenko V.I., Artyukhina E.V. Two Approaches to Training Radial Basis Neural Networks for Solving Partial Differential Equations. *Izvestiya Vuzov. Povolzhskiy Region. Technical Sciences. Informatics and Computer Technology*. 2007;2:56–66. (in Russ.).
6. Korsunov N.I., Lomakin A.V. Modelling Processes Described by the Wave Differential Equation Using Cellular Neural Networks. *Scientific Bulletins. Series History. Political Science. Economics. Informatics*. 2014;15(186):103–107. (in Russ.).
7. Kovalenko A.N., Chernomorets A.A., Petina M.A. On the Application of Neural Networks for Solving Partial Differential Equations. *Scientific Bulletins. Series Economics. Informatics*. 2017;258:103–110. (in Russ.).
8. Vershinin V.E., Ponomarev R.Y. Application of Neural Network Modelling Methods for Solving Initial-Boundary Value Problems for Partial Differential Equations. *Bulletin of Tyumen State University. Physical-Mathematical Modelling. Oil, Gas, Energy*. 2017;9(35):132–147. (in Russ.). <https://doi.org/10.21684/2411-7978-2023-9-3-132-147>
9. Zemskova Y.N. Applicability of Compactly Supported Neural Networks for Solving Partial Differential Equations Using the Finite Element Method. *Izvestiya PGPU im. V.G. Belinskogo*. 2009;13(17):144–148. (in Russ.).
10. Kansa E.J. Motivation for Using Radial Basis Functions to Solve PDEs. URL: <http://uahtitan.uah.edu/kansaweb.html> (accessed: January 16, 1999).
11. Kansa E.J. Multiquadrics. A Scattered Data Approximation Scheme with Applications to Computational Fluid Dynamics. II. Solutions to Parabolic, Hyperbolic, and Elliptic Partial Differential Equations. *Comput. Math. Appl.* 1990; 19(8/9):147–161.
12. Raissi M., Perdikaris P., Karniadakis G.E. Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations. *Journal of Computational Physics*. 2019;378:686–707.
13. Zrelova D.P., Ulyanov S.V. Models of Physically Informed Classical Lagrangian Hamiltonian Neural Networks in Deep Learning. *Modern Information Technologies and IT Education*. 2022;18(2):310–325. (in Russ.). <https://doi.org/10.25559/SITITO.18.202202.310-325>
14. Chen J., Viquerat J., Hachem E. U-net Architectures for Fast Prediction of Incompressible Laminar Flows. URL: <https://arxiv.org/pdf/1999.13532.pdf> (accessed: May 17, 1999).
15. Cai S., Wang Z., Wang S., Perdikaris P., Karniadakis G.E. Physics-Informed Neural Networks for Heat Transfer Problems. *Journal of Heat Transfer*. 2021;143(6):060801. <https://doi.org/10.1115/1.4050542>

Received 26.04.2024

Revised 14.05.2024

Accepted 15.05.2024

About the Author:

Alexander V. Galaburdin, associate professor of the department Mathematics and informatics, Don State Technical University (1, Gagarin Sq., Rostov-on-Don, 344003, RF), Cand.Sci. (Phys. – math.), associate professor, [ORCID](https://orcid.org/0000-0001-9151-1010), Galaburdin@mail.ru

Conflict of interest statement

The author does not have any conflict of interest.

The author has read and approved the final manuscript.

Поступила в редакцию 26.04.2024

Поступила после рецензирования 14.05.2024

Принята к публикации 15.05.2024

Об авторе:

Александр Васильевич Галабурдин, кандидат физико-математических наук, доцент кафедры математики и информатики Донского государственного технического университета (РФ, 344003, г. Ростов-на-Дону, пл. Гагарина, 1), [ORCID](#), Galaburdin@mail.ru

Конфликт интересов

Автор заявляет об отсутствии конфликта интересов

Автор прочитал и одобрил окончательный вариант рукописи.