

ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА COMPUTATIONAL MATHEMATICS



УДК 519.6

Оригинальное теоретическое исследование

<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>

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>

Введение. Для моделирования задач гидрогазодинамики требуется использовать всё более точные алгоритмы и подробные расчётные сетки, и, как следствие, большие вычислительные ресурсы, в том числе и методы распараллеливания расчётов на GPU ядра. Таким образом, необходима разработка нового решателя с адаптивным измельчением сеток на базе открытых платформ. Такой подход имеет ряд преимуществ по сравнению с написанием индивидуальных кодов. Во-первых, открытые платформы, как правило, предоставляют устоявшиеся и тщательно протестированные фреймворки, одобренные широким научным сообществом, что снижает риск ошибок и повышает общую надёжность. Во-вторых, использование открытых платформ способствует функциональной совместимости и возможности повторного использования, обеспечивая беспрепятственную интеграцию с другими инструментами и облегчая сотрудничество между исследователями. В-третьих, использование существующих открытых платформ позволяет значительно сократить время и стоимость разработки, поскольку эти платформы часто предлагают широкий спектр функциональных возможностей, начиная от обработки данных и заканчивая визуализацией и параллельными вычислениями. В-четвёртых, открытые платформы получают постоянное развитие и поддержку со стороны сообщества пользователей, что приводит к регулярным обновлениям, исправлению ошибок и повышению производительности. Это контрастирует с индивидуальными кодами, которые часто зависят исключительно от ресурсов и опыта человека или команды, создавшей их.

Среди свободного программного обеспечения был выбран AMReX, на сегодняшний день являющейся наиболее оптимальным открытым фреймворком, который позволяет:

- 1) использовать технологию адаптивного измельчения сеток (AMR);
- 2) использовать возможность распараллеливания вычислений на графические ядра GPU;
- 3) использовать метод погруженных границ при моделировании твёрдых тел в потоке;
- 4) строить структурированные сетки;
- 5) является с 2023 г. частью The High Performance Software Foundation, основанного Linux Foundation [1],

что обеспечивает огромную поддержку и дальнейшее развитие данного пакета программ.

На базе AMReX уже проводились численные расчёты задач газодинамики, в том числе и сравнение AMReX и OpenFOAM [2]. Поэтому было решено реализовать на базе AMReX решатель для задач гидро- и газодинами-

ки, использующий КГД уравнения. Аналогичный решатель уже реализован в OpenFOAM [3, 4] под общим названием QGDsolver [5], который показывает свою высокую эффективность. К сожалению, как уже отмечалось ранее, OpenFOAM не обладает возможностью распараллеливания вычислений на графические ядра и алгоритмом адаптивного измельчения сеток.

В представленной работе описана реализация квазигазодинамического (КГД) алгоритма в AMReX в упрощенной форме. В качестве упрощения выбран баротропный вариант системы уравнений газовой динамики, который позволяет избавиться от уравнения энергии и уравнения состояния. Этот баротропный вариант при известных предположениях имеет вид уравнений мелкой воды. Стоит отметить, что реализация решателя SWqgdAMR в программном комплексе AMReX является ключевым шагом в дальнейшем расширении набора решателей на базе КГД уравнений.

Сам КГД подход развивается на протяжении более 30 лет для задач газовой динамики и несжимаемых течений [6–10]. В последние годы КГД подход был реализован для задач в приближении мелкой воды [11–16].

Математическая модель и численный метод. Регуляризованные уравнения мелкой воды (РУМВ). Приведем РУМВ в векторной форме в отсутствии внешних сил и с условием плоского дна:

$$\frac{\partial h}{\partial t} + \nabla \cdot \mathbf{j}_m = 0, \quad (1)$$

$$\frac{\partial(h\mathbf{u})}{\partial t} + \nabla \cdot (\mathbf{j}_m \otimes \mathbf{u}) + \nabla \frac{gh^2}{2} = \nabla \Pi,$$

где h — толщина слоя воды; $\mathbf{j}_m = h(\mathbf{u} - \mathbf{w})$ — вектор плотности потока массы; \mathbf{u} — вектор скорости; g — ускорение силы тяжести; $\Pi = \Pi_{NS} + \Pi_{QGD}$ — тензор вязких напряжений; Π_{NS} — тензор вязких напряжений Навье-Стокса; \mathbf{w} , Π_{QGD} — КГД слагаемые; \otimes — векторное произведение. Здесь оператор набла, действующий на скаляр, означает градиент, на вектор — дивергенцию, на тензор — ковариантную производную: $\nabla T \equiv u|_{y_\beta} = \nabla_\alpha T_{\alpha\beta}$. Вид РУМВ с учетом формы дна и внешних сил есть, например, в [11–16].

Дискретизация регуляризованных уравнений мелкой воды.

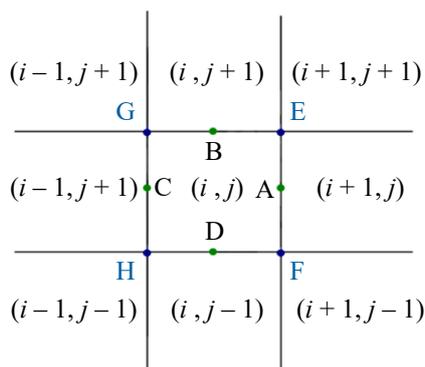


Рис. 1. Численный шаблон. Значения переменных h и \mathbf{u} отнесены к центрам ячеек с координатами (i, j)

КГД уравнения мелкой воды в покомпонентной форме имеют вид:

$$\begin{aligned} \frac{\partial h}{\partial t} + \frac{\partial j_{mx}}{\partial x} + \frac{\partial j_{my}}{\partial y} &= 0, \\ \frac{\partial hu_x}{\partial t} + \frac{\partial j_{mx}u_x}{\partial x} + \frac{\partial j_{my}u_x}{\partial y} + \frac{\partial}{\partial x} \left(\frac{gh^2}{2} \right) &= \frac{\partial \Pi_{xx}}{\partial x} + \frac{\partial \Pi_{yx}}{\partial y}, \\ \frac{\partial hu_y}{\partial t} + \frac{\partial j_{mx}u_y}{\partial x} + \frac{\partial j_{my}u_y}{\partial y} + \frac{\partial}{\partial y} \left(\frac{gh^2}{2} \right) &= \frac{\partial \Pi_{xy}}{\partial x} + \frac{\partial \Pi_{yy}}{\partial y}, \\ j_{mx} &= h(u_x - w_x), w_x = \frac{\tau}{h} \left(\frac{\partial(hu_x^2)}{\partial x} + \frac{\partial(hu_xu_y)}{\partial y} + gh \frac{\partial h}{\partial x} \right), \\ j_{my} &= h(u_y - w_y), w_y = \frac{\tau}{h} \left(\frac{\partial(hu_y^2)}{\partial y} + \frac{\partial(hu_xu_y)}{\partial x} + gh \frac{\partial h}{\partial y} \right), \\ \Pi_{xx} &= u_x \hat{w}_x + R + \Pi_{xx}^{NS}, \Pi_{xy} = u_x \hat{w}_y + \Pi_{xy}^{NS}, \\ \Pi_{yx} &= u_y \hat{w}_x + \Pi_{yx}^{NS}, \Pi_{yy} = u_y \hat{w}_y + R + \Pi_{yy}^{NS}, \\ \Pi_{xx}^{NS} &= h\nu \left(2 \frac{\partial u_x}{\partial x} - \frac{2}{3} \operatorname{div} \bar{\mathbf{u}} \right), \Pi_{yy}^{NS} = h\nu \left(2 \frac{\partial u_y}{\partial y} - \frac{2}{3} \operatorname{div} \bar{\mathbf{u}} \right), \end{aligned}$$

$$\begin{aligned}\Pi_{xy}^{NS} &= \Pi_{yx}^{NS} = h\nu \left(\frac{\partial u_y}{\partial x} + \frac{\partial u_x}{\partial y} \right), \quad \nu = \tau \frac{gh^2}{2}, \quad \operatorname{div} \bar{u} = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y}, \\ \hat{w}_x &= \tau \left(hu_x \frac{\partial u_x}{\partial x} + hu_y \frac{\partial u_x}{\partial y} + \frac{\partial}{\partial x} \left(\frac{gh^2}{2} \right) \right), \\ \hat{w}_y &= \tau \left(hu_y \frac{\partial u_y}{\partial y} + hu_x \frac{\partial u_y}{\partial x} + \frac{\partial}{\partial y} \left(\frac{gh^2}{2} \right) \right), \\ R &= g\tau \left(u_x \frac{\partial}{\partial x} \left(\frac{h^2}{2} \right) + u_y \frac{\partial}{\partial y} \left(\frac{h^2}{2} \right) + h^2 \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) \right).\end{aligned}$$

Запишем пространственную дискретизацию уравнений с учетом значений величин в полужелтых точках сетки (рис. 1):

$$\begin{aligned}h_A &= h_{i+\frac{1}{2},j} = 0,5(h_{i,j} + h_{i+1,j}), \quad h_B = h_{i,j+\frac{1}{2}} = 0,5(h_{i,j} + h_{i,j+1}), \\ h_C &= h_{i-\frac{1}{2},j} = 0,5(h_{i,j} + h_{i-1,j}), \quad h_D = h_{i,j-\frac{1}{2}} = 0,5(h_{i,j} + h_{i,j-1}), \\ h_E &= h_{i+\frac{1}{2},j+\frac{1}{2}} = 0,25(h_{i,j} + h_{i+1,j} + h_{i,j+1} + h_{i+1,j+1}), \\ h_F &= h_{i+\frac{1}{2},j-\frac{1}{2}} = 0,25(h_{i,j} + h_{i+1,j} + h_{i,j-1} + h_{i+1,j-1}), \\ h_G &= h_{i-\frac{1}{2},j+\frac{1}{2}} = 0,25(h_{i,j} + h_{i-1,j} + h_{i,j+1} + h_{i-1,j+1}), \\ h_H &= h_{i-\frac{1}{2},j-\frac{1}{2}} = 0,25(h_{i,j} + h_{i-1,j} + h_{i,j-1} + h_{i-1,j-1}).\end{aligned}$$

Аналогично записывается дискретизация компонент скорости в полужелтых точках. Дискретизация потока массы:

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

Дискретизация КГД слагаемых:

$$\begin{aligned}w_{xA} &= w_{xi+\frac{1}{2},j} = \frac{\tau}{h_{i+\frac{1}{2},j}} \left(\frac{h_{i+1,j} u_{xi+1,j} u_{xi+1,j} - h_{i,j} u_{xi,j} u_{xi,j}}{\Delta x} + \right. \\ &\quad \left. + \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} - h_{i+\frac{1}{2},j-\frac{1}{2}} u_{xi+\frac{1}{2},j-\frac{1}{2}} u_{xi+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 0,5g \frac{h_{i+1,j} h_{i+1,j} - h_{i,j} h_{i,j}}{\Delta x} \right), \\ w_{yB} &= w_{yi,j+\frac{1}{2}} = \frac{\tau}{h_{i,j+\frac{1}{2}}} \left(\frac{h_{i,j+1} u_{yi,j+1} u_{yi,j+1} - h_{i,j} u_{yi,j} u_{yi,j}}{\Delta y} + \right. \\ &\quad \left. + \frac{h_{i+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} u_{xi+\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} + 0,5g \frac{h_{i,j+1} h_{i,j+1} - h_{i,j} h_{i,j}}{\Delta y} \right), \\ w_{xC} &= w_{xi-\frac{1}{2},j} = \frac{\tau}{h_{i-\frac{1}{2},j}} \left(\frac{h_{i,j} u_{xi,j} u_{xi,j} - h_{i-1,j} u_{xi-1,j} u_{xi-1,j}}{\Delta x} + \right. \\ &\quad \left. + \frac{h_{i-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}} u_{xi-\frac{1}{2},j+\frac{1}{2}} - h_{i-\frac{1}{2},j-\frac{1}{2}} u_{xi-\frac{1}{2},j-\frac{1}{2}} u_{xi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y} + 0,5g \frac{h_{i,j} h_{i,j} - h_{i-1,j} h_{i-1,j}}{\Delta x} \right),\end{aligned}$$

$$\begin{aligned}
 w_{yD} &= w_{yi,j-\frac{1}{2}} = \frac{\tau}{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_{xD}^* &= 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_{xi+\frac{1}{2},j} \frac{u_{yi+1,j} - u_{yi,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. \\
 &+ \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_{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. \\
 &+ \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. \\
 &+ \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}$$

$$\begin{aligned}
 w_{yD}^* = w_{yi,j-\frac{1}{2}}^* = \tau \left(h_{i,j-\frac{1}{2}} \frac{u_{xi,j-\frac{1}{2}} - u_{xi,j-\frac{1}{2}}}{\Delta x} + h_{i,j-\frac{1}{2}} \frac{u_{yi,j-\frac{1}{2}} - u_{yi,j-\frac{1}{2}}}{\Delta y} + \right. \\
 \left. + 0,5g \frac{h_{i,j}h_{i,j} - h_{i,j-1}h_{i,j-1}}{\Delta y} \right), \\
 \operatorname{div} \bar{u}_A = \frac{u_{xi+1,j} - u_{xi,j}}{\Delta x} + \frac{u_{yi+\frac{1}{2},j+\frac{1}{2}} - u_{yi+\frac{1}{2},j-\frac{1}{2}}}{\Delta y}, \operatorname{div} \bar{u}_B = \frac{u_{yi,j+1} - u_{yi,j}}{\Delta y} + \frac{u_{xi+\frac{1}{2},j+\frac{1}{2}} - u_{xi-\frac{1}{2},j+\frac{1}{2}}}{\Delta x}, \\
 \operatorname{div} \bar{u}_C = \frac{u_{xi,j} - u_{xi-1,j}}{\Delta x} + \frac{u_{yi-\frac{1}{2},j+\frac{1}{2}} - u_{yi-\frac{1}{2},j-\frac{1}{2}}}{\Delta y}, \operatorname{div} \bar{u}_D = \frac{u_{yi,j} - u_{yi,j-1}}{\Delta y} + \frac{u_{xi+\frac{1}{2},j-\frac{1}{2}} - u_{xi-\frac{1}{2},j-\frac{1}{2}}}{\Delta x}.
 \end{aligned}$$

Параметр регуляризации алгоритма

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

вычисляется как

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

где α — настроечное число в пределах больше 0 и меньше 1; g — ускорение свободного падения. Аналогично записывается и дискретизация для τ_C , τ_D . Шаг по времени на базовой расчётной сетке выбирается так, чтобы удовлетворить условию устойчивости для явной схемы и имеет вид условия Куранта (число Куранта $0 < \beta < 1$):

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

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

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

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

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

Дискретизация тензора вязких напряжений:

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

Дискретизация уравнения сохранения массы:

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

Дискретизация уравнений баланса импульсов:

$$\begin{aligned} \hat{h}_{i,j}(\hat{u}_x)_{i,j} &- \frac{\Delta t}{\Delta x} \left((u_x)_{i+\frac{1}{2},j} (j_x)_{i+\frac{1}{2},j} - (u_x)_{i-\frac{1}{2},j} (j_x)_{i-\frac{1}{2},j} \right) - \frac{\Delta t}{\Delta y} \left((u_x)_{i,j+\frac{1}{2}} (j_y)_{i,j+\frac{1}{2}} - (u_x)_{i,j-\frac{1}{2}} (j_y)_{i,j-\frac{1}{2}} \right) - \\ &- 0,5g \frac{\Delta t}{\Delta x} \left(h_{i+\frac{1}{2},j} h_{i+\frac{1}{2},j} - h_{i-\frac{1}{2},j} h_{i-\frac{1}{2},j} \right) + \frac{\Delta t}{\Delta x} \left((\Pi_{xx})_{i+\frac{1}{2},j} - (\Pi_{xx})_{i-\frac{1}{2},j} \right) + \frac{\Delta t}{\Delta y} \left((\Pi_{yx})_{i,j+\frac{1}{2}} - (\Pi_{yx})_{i,j-\frac{1}{2}} \right), \\ \hat{h}_{i,j}(\hat{u}_y)_{i,j} &= h_{i,j}(u_y)_{i,j} - \frac{\Delta t}{\Delta y} \left((u_y)_{i,j+\frac{1}{2}} (j_y)_{i,j+\frac{1}{2}} - (u_y)_{i,j-\frac{1}{2}} (j_y)_{i,j-\frac{1}{2}} \right) - \\ &- \frac{\Delta t}{\Delta x} \left((u_y)_{i+\frac{1}{2},j} (j_x)_{i+\frac{1}{2},j} - (u_y)_{i-\frac{1}{2},j} (j_x)_{i-\frac{1}{2},j} \right) - 0,5g \frac{\Delta t}{\Delta y} \left(h_{i,j+\frac{1}{2}} h_{i,j+\frac{1}{2}} - h_{i,j-\frac{1}{2}} h_{i,j-\frac{1}{2}} \right) + \\ &+ \frac{\Delta t}{\Delta x} \left((\Pi_{yy})_{i,j+\frac{1}{2}} - (\Pi_{yy})_{i,j-\frac{1}{2}} \right) + \frac{\Delta t}{\Delta x} \left((\Pi_{xy})_{i+\frac{1}{2},j} - (\Pi_{xy})_{i-\frac{1}{2},j} \right). \end{aligned}$$

Реализация в AMReX. Численное решение уравнений мелкой воды реализовано на языке C++ с использованием открытого программного обеспечения AMReX. Данное программное обеспечение было взято за основу, поскольку позволяет использовать готовую логику дробления сеток по уровням, а также даёт возможность простого переноса (посредством макросов) вычислений на GPU ядра, что делает вычисления значительно менее затратными по времени.

На рис. 2 представлена структура разработанного программного обеспечения.

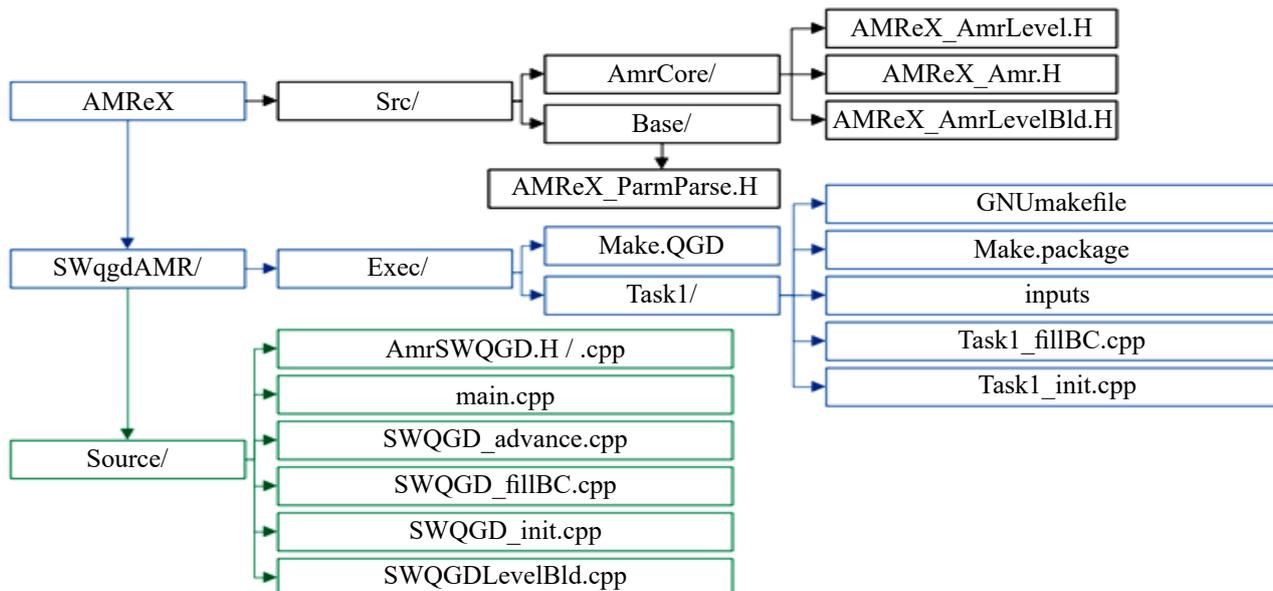


Рис. 2. Структура решателя

Основной класс-решатель AmrSWQGD объявлен в файле AmrSWQGD.H и реализован в файле AmrSWQGD.cpp. Он наследуется от класса AmrLevel, определенного в ядре AMReX. Наследование от этого класса позволяет простым способом измельчать сетку по уровням.

Логика итерации решателя определена в файле `SWQGD_advance.cpp` в методе `advance`. В нем при помощи цикла `ParallelFor` выполняется обновление полей h , u_x и u_y на каждой итерации. Здесь $huOld$ — контейнер старых переменных, $huNew$ — новых. Поскольку решение происходит на многоуровневой сетке (рис. 3), на каждом уровне которой своё разбиение (на уровне 0 самая грубая сетка, на уровнях выше точность увеличивается), то на каждом уровне вводится свой собственный шаг по времени (рис. 4). Например, если сетка имеет 2 уровня 0 и 1, и на уровне 1 сетка в 2 раза точнее по каждому направлению, чем на уровне 0, то одна итерация решения происходит следующим образом: происходит расчёт на уровне 0 с шагом по времени dt , происходят 2 итерации расчёта на уровне 1 с шагом по времени $dt/2$, а затем сетки синхронизируются. Такой алгоритм позволяет повысить точность вычислений.

Стоит отметить, что на каждом уровне необходимо уточнять не всю сетку, а только некоторые её части. Для этого класс-решатель определяет метод `errorEst` в файле `AmrSWQGD.cpp`. Данный метод принимает ссылку на экземпляр `tags` контейнера `TagBoxArray`. В цикле `ParallelFor` проходится каждая ячейка сетки, которая отмечается для разбиения, если подходит под некоторое условие (определяется в `if()` конструкции). Также для разбиения помечаются некоторые окружающие её ячейки. Остальные ячейки, которые не подходят под условие, помечаются меткой `clearval` и не будут подвергаться измельчению.

Сами расчётные задачи находятся в папке `Exes`, в этой папке содержатся файлы `inputs`, файл с начальными и граничными условиями.

В файле `inputs` записываются настройки решения:

- `max_step` — максимальное количество итераций;
- `stop_time` — расчётное время в секундах, на котором останавливается решение (по сути, расчёты идут до момента, пока количество итераций не превысит `max_step` или пока расчётное время не станет равным `stop_time`);
- `geometry.is_periodic` — массив из 3 булевых переменных (например, 0 0 0 или 0 1 1 или 1 0 1), определяющий будут ли границы по определённому направлению периодичны (1) или нет (0);
- `geometry.coord_sys` — система координат, в которой происходит решение (рекомендуемое значение 0, что соответствует декартовой системе координат; гарантии того, что решатель будет корректно работать в других системах, не даются);
- `geometry.prob_lo` — хуз координаты левого нижнего угла физической прямоугольной области (например, 0.0 0.0 0.0);
- `geometry.prob_hi` — хуз координаты правого верхнего угла физической области (например, 10. 10. 1.);
- `amr.n_cell` — массив из трёх целых переменных, обозначающих разбиение сетки в каждом из направлений на 0 уровне (например, 512 512 1);
- `amr.max_level` — целое число, обозначающее максимальный допустимый уровень разбиения сетки;
- `amr.ref_ratio` — коэффициент дробления сетки по уровням;
- `amr.regrid_int` — целое число, обозначающее количество шагов, через которое следует пересоздать сетку;
- `amr.max_grid_size` — решатель делит сетку на домены, размер каждого не превосходит `max_grid_size` по каждому из направлений;
- `amr.plot_int` — решатель записывает в файлы каждые `plot_int` итераций.

Уравнения граничных условий описаны в структуре `SWQGDBCFill` в файле `Task1_fillBC.cpp` (`Task1` — название задачи, может быть изменено на любое другое). Здесь `ilo` — левая граница расчётной области, `ihi` — правая граница, `jlo` — нижняя граница, `jhi` — верхняя граница. Уравнения решаются для переменных h , u_x и u_y , т. е. высоты столба воды, скорости по x -направлению и скорости по y -направлению. Переменная `dest` представляет собой многомерный массив, хранящий значения всех переменных во всей расчётной области. Обратиться к переменной h в ячейке (i, j, k) можно с помощью `dest(i, j, k, 0)`, к переменной u_x в той же ячейке с помощью `dest(i, j, k, 1)`, а к переменной u_y , соответственно, как `dest(i, j, k, 2)`. Так как у нас 2D уравнения, то индекс k в данном случае полагается равным 1.

Начальные условия задаются в файле `Task1_init.cpp`. Здесь переменная `snew[bi]` выступает в роли контейнера, аналогичного контейнеру `dest` в структуре для граничных условий. Начальные условия рассчитываются в цикле `amrex::ParallelFor`. Данный цикл вместе с макросом — лямбда-функцией `AMREX_GPU_DEVICE` позволяет выполнить расчёт граничных условий во всей области параллельно на GPU ядрах.

Для компиляции программы необходимо перейти в папку с задачей (`Task1`), затем в терминале запустить команду `make -j n`, где n — количество ядер для распараллеливания. Для запуска утилиты без распараллеливания достаточно запустить просто `make`. После завершения работы `make` в папке появится файл `main2d.gnu.MPI.ex` (название может немного отличаться). Далее для запуска необходимо в терминале прописать следующую команду: `mpirun -np n ./main2d.gnu.MPI.ex inputs`, где n — количество ядер для распараллеливания, после чего запустится расчёт.

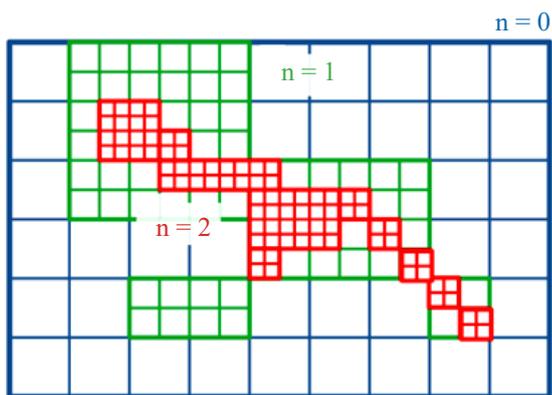


Рис. 3. Схема алгоритма адаптивного измельчения сетки

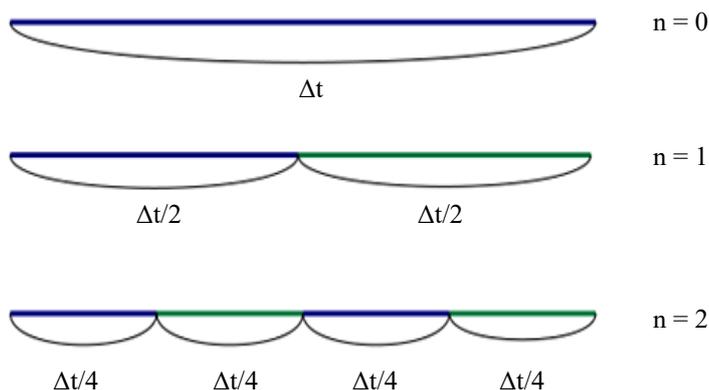


Рис. 4. Схема алгоритма дробления временного шага в подцикле

Результаты исследования. Для валидации и верификации разработанного решателя используются две 2D задачи: 1) задача о разрушении радиальной плотины, для которой хорошо известно аналитическое решение, 2) задача о распаде двух разных по высоте столбов жидкости.

Двумерная задача о разрушении радиальной плотины (Circular Dam-Break). Задача о распаде столба жидкости или о прорыве круглой дамбы (Circular Dam-Break) является широко распространённой при валидации и верификации новых решателей [17–20]. Рассматривается 2D плоскость размером 40×40 м, в центре которой находится столб жидкости высотой $h = 2,5$ м и с радиусом $R = 2,5$ м, высота жидкости в остальной области составляет $h_0 = 0,5$ м (рис. 5). Расчётная область разбивается на 40 000 однородных ячеек, то есть по 200 в каждом направлении. Шаг по времени выбирается равным $\Delta t = 10^{-4}$ с, расчёт проводился до времени $t = 4,7$ с.

Визуализация на рис. 6 показывает распад столба жидкости. В начальный момент времени убирается стенка и наблюдается движение воды во всех направлениях. Затем, в то время как круговая ударная волна распространяется наружу, внутрь исходного цилиндра движется волна разрежения до тех пор, пока она полностью не сходится в центре расчётной области, где она отражается, что вызывает градиент высоты, а как следствие и вторичную ударную волну. На рис. 7 приведены результаты численных экспериментов в сравнении с аналитическим решением из [19] в момент времени $t = 4,7$ с. Показана (рис. 7 а) зависимость решения от параметра настройки алгоритма α . Оптимальным значением является $\alpha = 0,2$. Сходимость решения по сетке показана на рис. 7 б. Характерное число Куранта равно 0,2.

Пример работы алгоритма адаптивного измельчения сетки можно видеть на рис. 8. Здесь в зависимости от выбранного критерия адаптации (в нашем случае это градиент высоты столба жидкости) выполняется дробление сетки по уровням (в нашем случае $n_{amr} = 4$, где n_{am} — максимальный уровень в текущем расчёте), что значительно ускоряет расчёт. Детальное исследование приведено в разделе об исследовании производительности решателя SWqgdAMR.

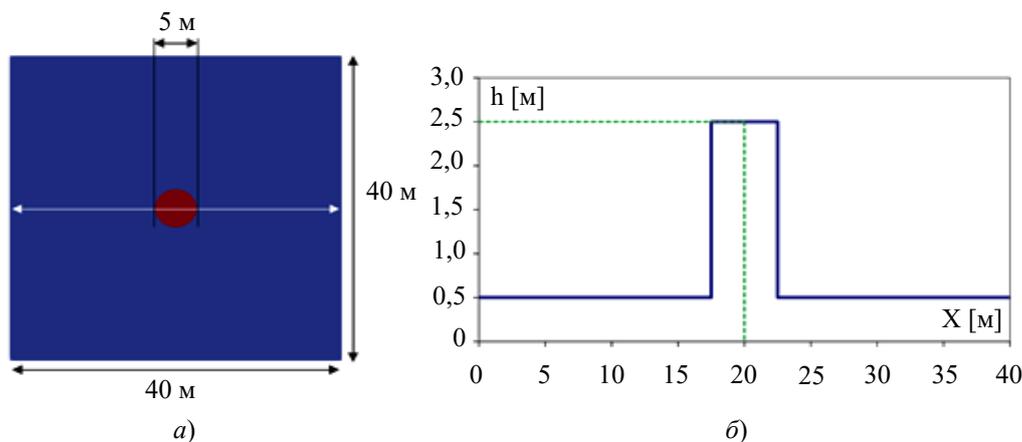


Рис. 5. Начальные условия задачи о разрушении радиальной плотины: а — геометрия расчётной области и начальное распределение; б — высота столба жидкости вдоль белой линии

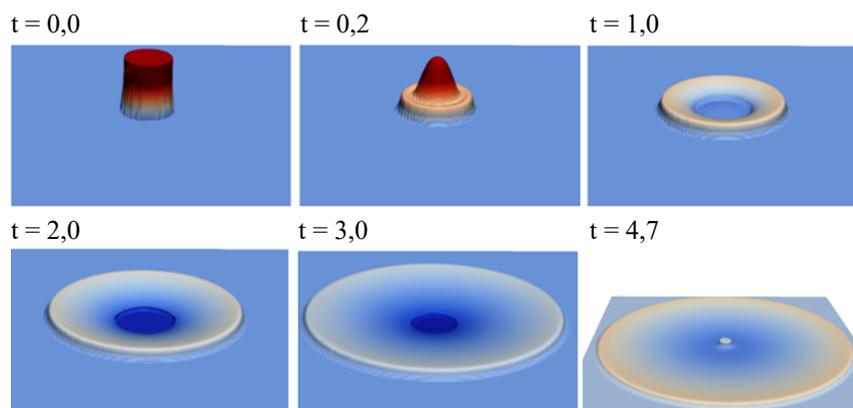


Рис. 6. Визуализация распада столба жидкости со временем $\alpha = 0,2$, $\Delta t = 10^{-4}$ с

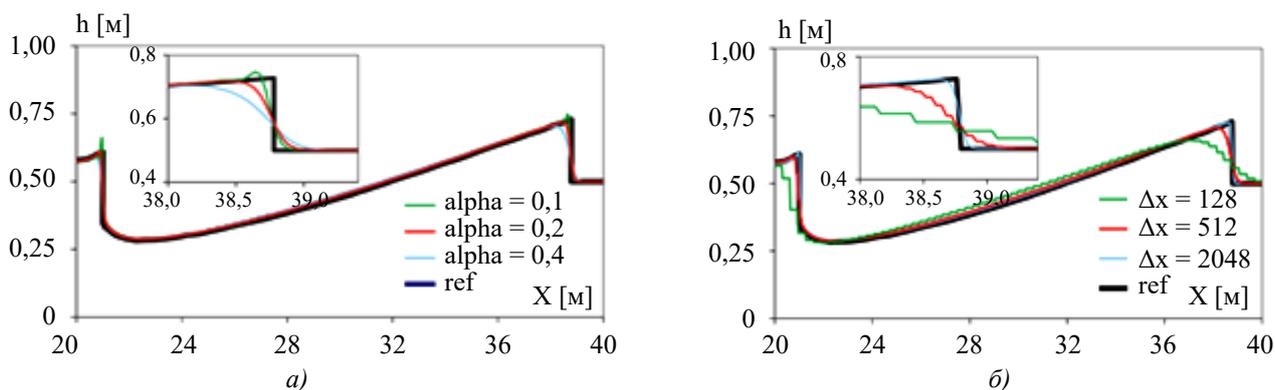


Рис. 7. Распределение высоты столба жидкости с шагом по времени $\Delta t = 10^{-4}$ с в момент времени $t = 4,7$ с:
 а — зависимость от параметра α на постоянной сетке $\Delta x = 1024$;
 б — сходимость по сетке при постоянном $\alpha = 0,2$

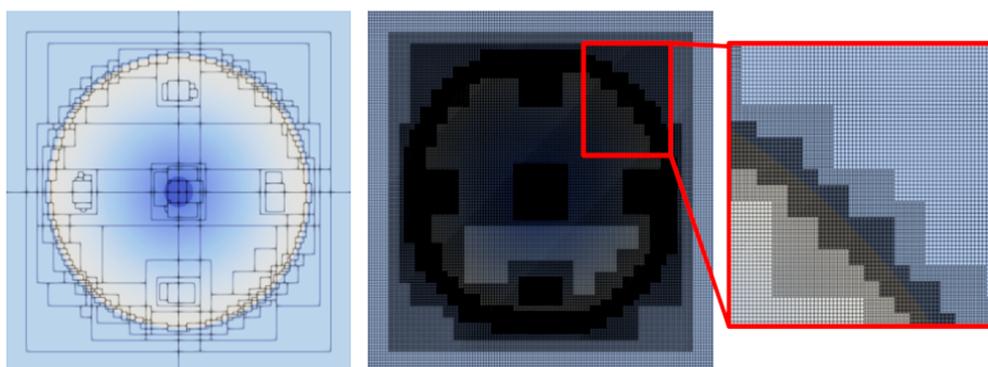


Рис. 8. Визуализация работы алгоритма адаптивного измельчения сетки при $\alpha = 0,2$ в момент времени $t = 3$ с.
 На первом рисунке изображено разбиение на блоки, на втором — адаптивная сетка

Распад двух разных по высоте столбов жидкости. Как и предыдущий тест, задача о распаде двух разных по высоте столбов жидкости позволяет проверить возможность решателя воспроизводить сложные структуры течения. Рассматривается 2D плоскость размером 2000×2000 м, в точке $(875,0)$ располагается первый столб воды радиуса $R_1 = 125$ м и высотой $h_1 = 15$ м, в точке с координатами $(1375,0)$ располагается второй столб воды $R_2 = 125$ м, $h_2 = 20$ м, уровень воды в остальной области равен $h_1 = 10$ м (рис. 9). Расчётная область разбивается на 160 000 однородных ячеек, то есть по 400 в каждом направлении. Шаг по времени выбирается равным $\Delta t = 10^{-4}$ с, расчёт заканчивался в момент времени $t = 30$ с.

Визуализация на рис. 10 и 11 показывает распад и дальнейшее взаимодействие двух столбов жидкости. В начальный момент времени убираются стенки и наблюдается движение воды от каждого столба во всех направлениях. Затем две ударные волны сталкиваются друг с другом и происходит сильная деформация фронта двух волн.

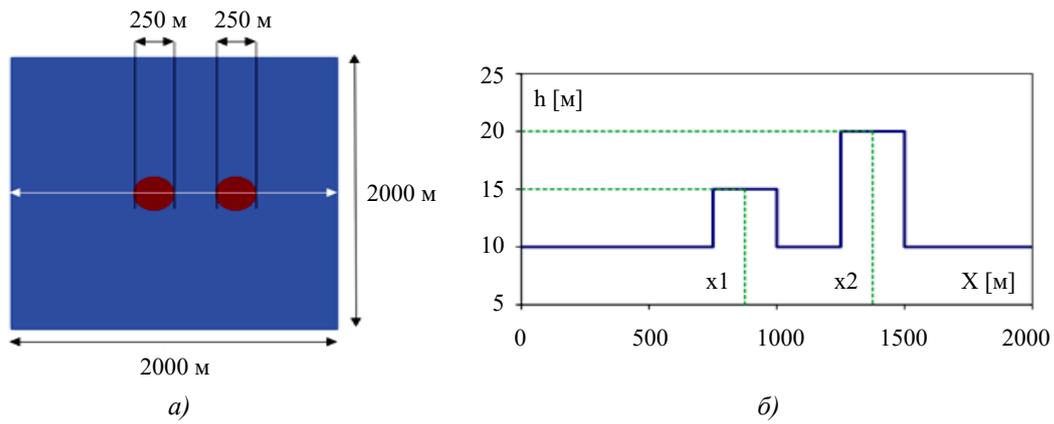


Рис. 9. Начальные условия задачи о распаде двух разных по высоте столбов жидкости:
 а — геометрия расчётной области;
 б — высота столба жидкости вдоль белой линии, $x_1 = 875$ м, $x_2 = 1375$ м

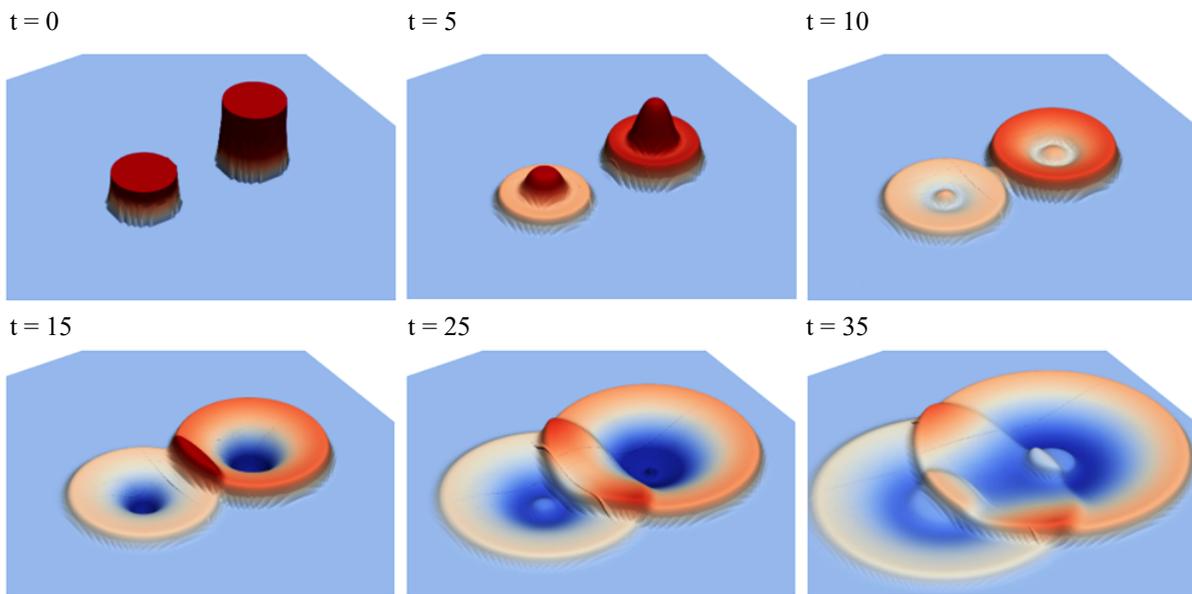


Рис. 10. Визуализация распада двух столбов жидкости со временем $\alpha = 0,2, \Delta t = 10^{-4}$ с.
 Время на рисунке в секундах

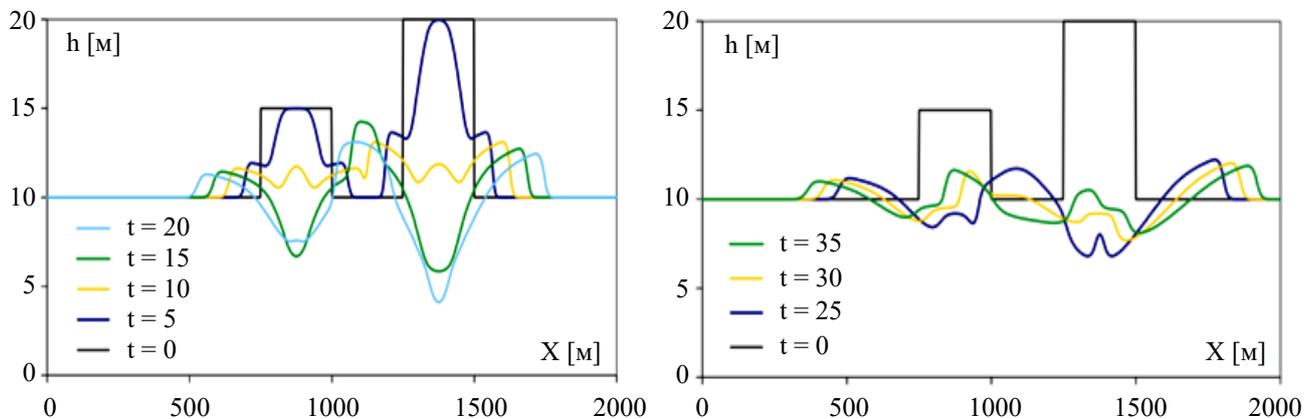


Рис. 11. График распада двух столбов жидкости со временем в центральном сечении.
 Время на рисунке в секундах, $\Delta x = 1024, \alpha = 0,2, \Delta t = 10^{-4}$ с

Производительность решателя SWqgdAMR. Одним из важных критериев при разработке нового решателя является оценка его производительности и эффективности распараллеливания. Для этого была использована задача о распаде двух разных по высоте столбов жидкости. Было взято 1 048 576 ячеек расчётной сетки, шаг по времени

$\Delta t = 10^{-4}$ с, расчёт завершился при $t = 0,1$ с. Для оценки производительности решателя использовался 1 физический процессор Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz с 8 ядрами, результаты представлены в таблице 1.

Таблица 1

Оценка производительности решателя SWqgdAMR

Количество ядер	Количество ячеек	Время расчёта t , с	Эффективность, %
1	1 048 576	71	–
2	524 288	40	89
4	262 144	27	66

На том же процессоре с использованием 2 ядер было исследовано ускорение вычислений за счёт использования технологии адаптивного измельчения сетки. Данные представлены в таблице 2. Количество расчётных ячеек при $n_{\text{amr}} = 0$ равно 4 194 304, базовая сетка при $n_{\text{amr}} = 1$ равна 1 048 576, при $n_{\text{amr}} = 2$ равно 262 144, при $n_{\text{amr}} = 4$ базовая расчётная сетка 65 536, при этом во всех случаях разрешение течения остаётся постоянным, качество может отличаться в зависимости от критерия адаптации (пример работы алгоритма измельчения демонстрирует рис. 8). В наших расчётах в качестве критерия адаптации сетки был выбран градиент высоты столба жидкости.

Таблица 2

Время вычисления в секундах в зависимости от уровней адаптации сетки

Количество ядер	Количество ячеек	$n_{\text{amr}} = 0$	$n_{\text{amr}} = 1$	$n_{\text{amr}} = 2$	$n_{\text{amr}} = 4$
2	4 194 304	2288	359	132	41

Таким образом, использование технологии адаптивного измельчения сетки позволяет значительно ускорять вычисления. В нашем случае достигается ускорение до 56 раз. В работе [21] показано, что на одинаковых стационарных сетках AMReX быстрее OpenFOAM в 4 раза, таким образом с использованием измельчения сетки можно добиться выигрыша в скорости до 232 раз.

Обсуждение и заключения. В работе 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.

Решатель SWqgdAMR написан в рамках работы по расширению 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. Следующий этап работ включает в себя добавление в AMReX квазигазодинамических уравнений для расчёта задач газодинамики.

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

Список литературы

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> (дата обращения: 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> (дата обращения: 16.04.2024).
6. Elizarova T.G. *Quasi-gas-dynamic Equations*. Springer. 2009.
7. Елизарова Т.Г. *Квазигазодинамические уравнения и методы расчета вязких течений*. Москва: Научный мир; 2007. 350 с.

8. Четверушкин Б.Н. *Кинетические схемы и квазигазодинамическая система уравнений*. Москва: МАКС Пресс; 2004. 328 с.
9. Шеретов Ю.В. *Регуляризованные уравнения гидродинамики*. Тверь: Тверской государственный университет; 2016. 222 с.
10. Шеретов Ю.В. *Динамика сплошных сред при пространственно-временном осреднении*. Москва; Ижевск: Регулярная и хаотическая динамика; 2009. 400 с.
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. Булатов О.В., Елизарова Т.Г. Регуляризованные уравнения мелкой воды и эффективный метод численного моделирования течений в неглубоких водоемах. *Журнал вычислительной математики и математической физики*. 2011;51(1):170–184.
15. Елизарова Т.Г., Иванов А.В. Регуляризованные уравнения для численного моделирования течений в приближении двухслойной мелкой воды. *Журнал вычислительной математики и математической физики*. 2018;58(5):741–761.
16. Иванов А.В. О реализации модели мелкой воды на базе квазигазодинамического подхода в открытом программном комплексе OpenFOAM. *Препринты Института прикладной математики им. М.В. Келдыша РАН*. 2023;28:27.
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-Fraão 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.

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: MAKS Press; 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.

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](https://orcid.org/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](https://orcid.org/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](https://orcid.org/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](https://orcid.org/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](https://orcid.org/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.

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, Miuskaya 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 Koptyuga 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, Miuskaya 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.