



doi 10.26089/NumMet.v26r102

УДК 519.673

Декомпозиция расчетной области для численного решения уравнения Кана–Хиллиарда

Д. И. Прохоров

Институт нефтегазовой геологии и геофизики имени А. А. Трофимука СО РАН,
Новосибирск, Российская Федерация

ORCID: 0000-0002-8547-930X, e-mail: prokhorovdi@ipgg.sbras.ru

Аннотация: В статье представлен параллельный алгоритм для численного решения уравнения Кана–Хиллиарда в трехмерной постановке. В основе алгоритма лежит явная конечно-разностная схема и метод декомпозиции расчетной области. Для каждой подобласти выделяется один MPI-процесс, при этом решение внутри подобласти вычисляется с использованием технологии CUDA для GPU. Исследована сильная и слабая масштабируемость алгоритма в приложении к задаче разделения фаз для двух различных схем обмена данными между RAM и VRAM.

Ключевые слова: метод фазового поля, декомпозиция расчетной области, уравнение Кана–Хиллиарда, гибридные вычисления.

Благодарности: Исследование выполнено за счет гранта Российского научного фонда № 24–71–00028.

Для цитирования: Прохоров Д.И. Декомпозиция по пространственным подобластям для численного решения уравнения Кана–Хиллиарда // Вычислительные методы и программирование. 2025. **26**, № 1. 17–32. doi 10.26089/NumMet.v26r102.

Domain decomposition for the numerical solution of the Cahn–Hilliard equation

Dmitry I. Prokhorov

Trofimuk Institute of Petroleum Geology and Geophysics SB RAS, Novosibirsk, Russia

ORCID: 0000-0002-8547-930X, e-mail: prokhorovdi@ipgg.sbras.ru

Abstract: The article presents a parallel algorithm for the numerical solution of the Cahn–Hilliard equation in a three-dimensional formulation. The algorithm is based on an explicit finite difference scheme and a domain decomposition method. One MPI process is allocated for each subdomain, while the solution within the subdomain is calculated using CUDA technology for the GPU. The strong and weak scaling of the algorithm is investigated in application to the problem of phase separation for two different data exchange schemes between RAM and VRAM.

Keywords: phase-field method, domain decomposition, Cahn–Hilliard equation, heterogeneous computing.

Acknowledgements: This work was supported by the Russian Science Foundation under grant No. 24–71–00028.

For citation: D. I. Prokhorov, “Domain decomposition for the numerical solution of the Cahn–Hilliard equation,” Numerical Methods and Programming. **26** (1), 17–32 (2025). doi 10.26089/NumMet.v26r102.



1. Введение. Многофазные процессы в масштабе пор находят важное применение в таких областях, как повышение нефтеотдачи пластов [1], улавливание и секвестрация CO_2 [2, 3] и разработка новых материалов [4, 5]. Например, распространенным методом увеличения нефтеотдачи пластов в нефтяной промышленности является заводнение, поэтому один из типичных сценариев многофазного течения в пористой среде — движение воды и нефти в пласте-коллекторе. В процессе заводнения в поровом пространстве образуются кластеры остаточной нефти, что снижает скорость извлечения. При этом распределение остаточной нефти зависит от свойств потока жидкости [6].

Однако продолжительное заводнение не оказывает значительного влияния на нефтеотдачу [7]. Увеличение нефтеотдачи высоководных пластов может быть достигнуто за счет вытеснения газом [8]. Помимо увеличения нефтеотдачи, закачка газа имеет дополнительные преимущества. Например, закачка CO_2 способствует сокращению углеродных выбросов [9], а закачка углеводородного газа может использоваться для регулирования пиков добычи природного газа в разные сезоны. Однако неоднородность пласта и различие вязкостей нефти, газа и воды влияют на эффективность циркуляции газа в пласте, а следовательно, на нефтеотдачу пласта и его емкость для хранения газа [10]. Поэтому изучение механизмов многофазных течений в пористой среде необходимо для разработки эффективной программы заводнения или закачки газа с целью увеличения нефтеотдачи пластов. Другим примером является спекание зерен различных материалов с различной кристаллографической ориентацией, которые обычно возникают в области производства порошковой металлургии и керамики [11].

Основной сложностью при решении многофазных задач является аппроксимация граничных условий на движущейся границе фаз. Например, для решения уравнения Навье–Стокса, описывающего многофазное течение в открытых порах, применяются сеточные методы (метод конечных объемов (FVM) [12], метод конечных разностей (FDM) [13], метод конечных элементов (FEM) [14]) совместно с методами неявного отслеживания межфазной границы (interface-capturing): метод объема жидкости (VOF) [15, 16], метод поверхности уровня (LS) [17, 18] и метод фазового поля (PF) [19, 20]. Также существует ряд методов, опирающихся на альтернативные подходы: решеточный метод Больцмана (LBM) [21], метод сглаженных частиц [22], модель поровой сети [23].

Среди методов неявного отслеживания межфазной границы отдельно стоит выделить метод фазового поля [20]. Этот метод изначально разработан для моделирования разделения фаз [24]. Однако в дальнейшем он получил также широкое распространение в задачах моделирования спекания и затвердевания [5, 25]. Основная идея метода заключается в использовании консервативного параметра порядка, например массовой доли, который непрерывно изменяется на границе фаз и практически постоянен на удалении от границы, т.е. резкая граница раздела заменяется тонким, но ненулевым по толщине переходным слоем, где поверхностные силы заданы гладкими функциями [20]. Важной особенностью моделей фазового поля является то, что они учитывают законы сохранения и термодинамические принципы [26, 27], а также позволяют задать граничные условия на углы контакта [28]. По сравнению с методами явного отслеживания фронта [29, 30], метод фазового поля позволяет обрабатывать сложные топологические изменения на границе раздела фаз.

Большинство моделей фазового поля основаны на уравнении Кана–Хиллиарда [31], которое является нелинейным и имеет четвертый порядок. Решение такого уравнения представляет собой сложную вычислительную задачу. Более того, при решении прикладных задач возникает необходимость использования для численного решения уравнения Кана–Хиллиарда достаточно больших вычислительных сеток, размер которых может существенно превышать 1000^3 узлов. В задачах моделирования потока на масштабе пор большой размер задачи обусловлен репрезентативным размером цифровых изображений горных пород [32, 33], а в задачах спекания — ростом зерен [11].

Важно отметить, что многие работы, описывающие численное моделирование в многофазных средах, ограничиваются рассмотрением двумерного случая, а в работах, посвященных трехмерной постановке, вычислительные эксперименты проводятся на небольших вычислительных сетках размером 300^3 узлов. В исследовании [34] проведены эксперименты по моделированию двухфазных потоков на вычислительных сетках размером более 1000^3 узлов с использованием графических ускорителей. Для разделения задач между графическими ускорителями используется одномерная (slab) и двумерная (pencil) декомпозиция расчетной области.

В данной работе для решения уравнения Кана–Хиллиарда также используются гибридные вычисления. Разработанный алгоритм основан на явной конечно-разностной схеме и трехмерной декомпозиции пространственной области, где для каждой пространственной подобласти назначается один MPI-процесс.



Распараллеливание подзадач выполняется с использованием технологии CUDA для графических ускорителей.

Статья организована следующим образом: модель описана в разделе 2, соответствующая конечно-разностная схема представлена в разделе 3, в разделе 4 описано распараллеливание алгоритма, анализ производительности представлен в разделе 5.

2. Постановка задачи. Рассматривается двухфазная система в прямоугольной области $\Omega = [X_1^{\min}, X_1^{\max}] \times [X_2^{\min}, X_2^{\max}] \times [X_3^{\min}, X_3^{\max}] \subset \mathbb{R}^3$. Пространственное распределение фаз определяется концентрацией $0 \leq c \leq 1$ одной из фаз.

В основе уравнения Кана–Хиллиарда лежит потенциал свободной энергии Гинзбурга–Ландау [31]:

$$F = \int f(c) + \frac{1}{2}\beta|\nabla c|^2 d\Omega,$$

где $f(c) = Ac^2(1-c)^2$ — плотность свободной энергии однородной системы, β и A — параметры, определяющие поверхностную энергию. Функциональная производная потенциала F — химический потенциал

$$\mu = \frac{\delta F}{\delta c} = \frac{\partial f}{\partial c} - \beta\Delta c.$$

Поток концентрации пропорционален градиенту химического потенциала μ :

$$J = -D(c)\nabla\mu, \tag{1}$$

где $D(c)$ — коэффициент диффузии. Форма коэффициента диффузии зависит от рассматриваемой задачи. В задаче о двухфазном потоке несжимаемых жидкостей метод фазового поля используется для отслеживания интерфейса и коэффициент диффузии считается постоянным [35]. В случае моделирования спекания коэффициент диффузии описывает различные механизмы массопереноса [25]:

$$D = D_{\text{vol}}\phi(c) + D_{\text{vap}}(1 - \phi(c)) + D_{\text{surf}}c^2(1 - c^2), \tag{2}$$

где

$$\phi(c) = c^3(10 - 15c + 6c^2),$$

$D_{\text{vol}}, D_{\text{vap}}, D_{\text{surf}}$ — коэффициенты объемной диффузии, диффузии испарения и поверхностной диффузии соответственно. Коэффициент диффузии D так же может включать в себя дополнительные члены, например связанные с диффузией на границе зерен.

Поскольку система замкнута, суммарная концентрация не изменяется. Таким образом, с использованием уравнения неразрывности выражение (1) преобразовывается в уравнение Кана–Хиллиарда:

$$\frac{\partial c}{\partial t} = -\nabla \cdot J.$$

На границе области Ω может быть задано условие Дирихле или Неймана, также граничное условие может определяться контактным углом. Для удобства далее рассматриваются периодические граничные условия.

3. Конечно-разностная схема. Для численного решения уравнения Кана–Хиллиарда используется явная конечно-разностная схема на сдвинутых сетках с шагами дискретизации по пространству h_m , $m = 1, 2, 3$ [36]. Будем использовать целочисленные узлы $(x_m)_j = h_m j$ и полуцелочисленные узлы $(x_m)_{j+1/2} = h_m(j + 1/2)$. Концентрация c , химический потенциал μ и коэффициент диффузии D определены в узлах с целочисленными координатами, а компоненты потока концентрации J определены в узлах, одна из координат которых является полуцелочисленной: $(J_1)_{j_1+1/2j_2j_3}$, $(J_2)_{j_1j_2+1/2j_3}$, $(J_3)_{j_1j_2j_3+1/2}$.

Введем обозначения конечно-разностных операторов для аппроксимации первой и второй производной по соответствующему направлению

$$D_{x_1}^1 [g]_{j_1j_2j_3} = \frac{g_{j_1+1/2j_2j_3} - g_{j_1-1/2j_2j_3}}{h_1} = \frac{\partial g}{\partial x_1} \Big|_{j_1j_2j_3} + O(h_1^2),$$

$$D_{x_1}^2 [g]_{j_1j_2j_3} = \frac{g_{j_1+1j_2j_3} - 2g_{j_1j_2j_3} + g_{j_1-1j_2j_3}}{h_1^2} = \frac{\partial^2 g}{\partial x_1^2} \Big|_{j_1j_2j_3} + O(h_1^2),$$

оператора усреднения по соответствующему направлению

$$\mathbf{A}_{x_1}[g]_{j_1 j_2 j_3} = \frac{g_{j_1+1/2 j_2 j_3} + g_{j_1-1/2 j_2 j_3}}{2}$$

и конечно-разностного оператора Лапласа

$$\mathbf{L}[g]_{j_1 j_2 j_3} = \mathbf{D}_{x_1}^2[g]_{j_1 j_2 j_3} + \mathbf{D}_{x_2}^2[g]_{j_1 j_2 j_3} + \mathbf{D}_{x_3}^2[g]_{j_1 j_2 j_3}.$$

Используем следующую разностную схему:

$$\frac{c_{j_1 j_2 j_3}^{n+1} - c_{j_1 j_2 j_3}^n}{\tau} = -(\mathbf{D}_{x_1}^1[J_1]_{j_1 j_2 j_3} + \mathbf{D}_{x_2}^1[J_2]_{j_1 j_2 j_3} + \mathbf{D}_{x_3}^1[J_3]_{j_1 j_2 j_3}), \quad (3)$$

$$\begin{aligned} (J_1)_{j_1+1/2 j_2 j_3}^n &= -\mathbf{A}_{x_1}[\mathbf{D}]_{j_1+1/2 j_2 j_3}^n \mathbf{D}_{x_1}^1[\mu]_{j_1+1/2 j_2 j_3}, \\ (J_2)_{j_1 j_2+1/2 j_3}^n &= -\mathbf{A}_{x_2}[\mathbf{D}]_{j_1 j_2+1/2 j_3}^n \mathbf{D}_{x_2}^1[\mu]_{j_1 j_2+1/2 j_3}, \\ (J_3)_{j_1 j_2 j_3+1/2}^n &= -\mathbf{A}_{x_3}[\mathbf{D}]_{j_1 j_2 j_3+1/2}^n \mathbf{D}_{x_3}^1[\mu]_{j_1 j_2 j_3+1/2}, \end{aligned} \quad (4)$$

$$\mu_{j_1 j_2 j_3}^n = \frac{\partial f}{\partial c}(c_{j_1 j_2 j_3}^n) - \beta \mathbf{L}[c]_{j_1 j_2 j_3}^n. \quad (5)$$

Оператор $\mathbf{D}_{x_1}^1$ применяется как в узлах с целочисленными координатами (3), так и в узлах, одна из координат которых является полуцелочисленной (4).

Зададим периодические граничные условия

$$\begin{aligned} (x_m)_{j_m^{\min}-1} &= (x_m)_{j_m^{\max}}, \\ (x_m)_{j_m^{\max}+1} &= (x_m)_{j_m^{\min}}, \end{aligned}$$

где $j_m^{\min} h_m = X_m^{\min}$, $(j_m^{\max} + 1) h_m = X_m^{\max}$.

4. Параллельная реализация алгоритма. Параллельная реализация конечно-разностной схемы выполнена с использованием метода декомпозиции расчетной области на языке C++. Метод декомпозиции расчетной области заключается в ее разбиении на подобласти, не пересекающиеся друг с другом. Вычисления в каждой из подобластей проводятся независимо друг от друга. В данной работе рассматриваются подобласти, представляющие собой прямоугольные параллелепипеды. К границе каждой подобласти добавляются два слоя узлов, соответствующих узлам вычислительной сетки из соседних подобластей. На каждой итерации конечно-разностной схемы новое распределение концентрации вычисляется внутри подобласти, а затем значения в дополнительных слоях узлов обновляются соответствующими данными из соседних подобластей. Для того чтобы расчеты в подобластях проводились параллельно, каждой из подобластей назначается один MPI-процесс, при этом конечно-разностные вычисления внутри подобласти распараллелены с использованием технологии CUDA для графических ускорителей. Для минимизации объема данных, участвующих в MPI-обменах, вычислительная область разбивается по всем трем направлениям; таким образом, подобласти имеют дополнительные слои узлов с каждой из шести сторон.

Поскольку MPI-обмены между процессами и обмены данными между RAM и VRAM увеличивают время работы программы, естественным является предложение организовать расчеты так, чтобы часть расчетов выполнялась параллельно с обменами данными. Далее представлены два способа такой организации вычислений.

4.1. Подпрограмма для решения уравнения Кана–Хиллиарда в отдельной подобласти.

Структура программы для расчетов в отдельной подобласти с использованием библиотеки CUDA представлена на рис. 1. Вначале исходное распределение концентрации c^0 загружается в память GPU. Затем на каждой итерации последовательно вычисляются μ , D , J , c^{n+1} и происходит переписывание переменных, соответствующих конечно-разностным сеткам для c^n и c^{n+1} . После выполнения заданного числа итераций массив с итоговым распределением концентрации перемещается в RAM. Поскольку на границе заданы периодические граничные условия, дополнительные слои ячеек не добавляются, так как программа имеет доступ к узлам на противоположной границе области напрямую. Далее последовательное вычисление параметров μ , D , J , c^{n+1} в некоторой области обозначается как подпрограмма конечно-разностных вычислений.

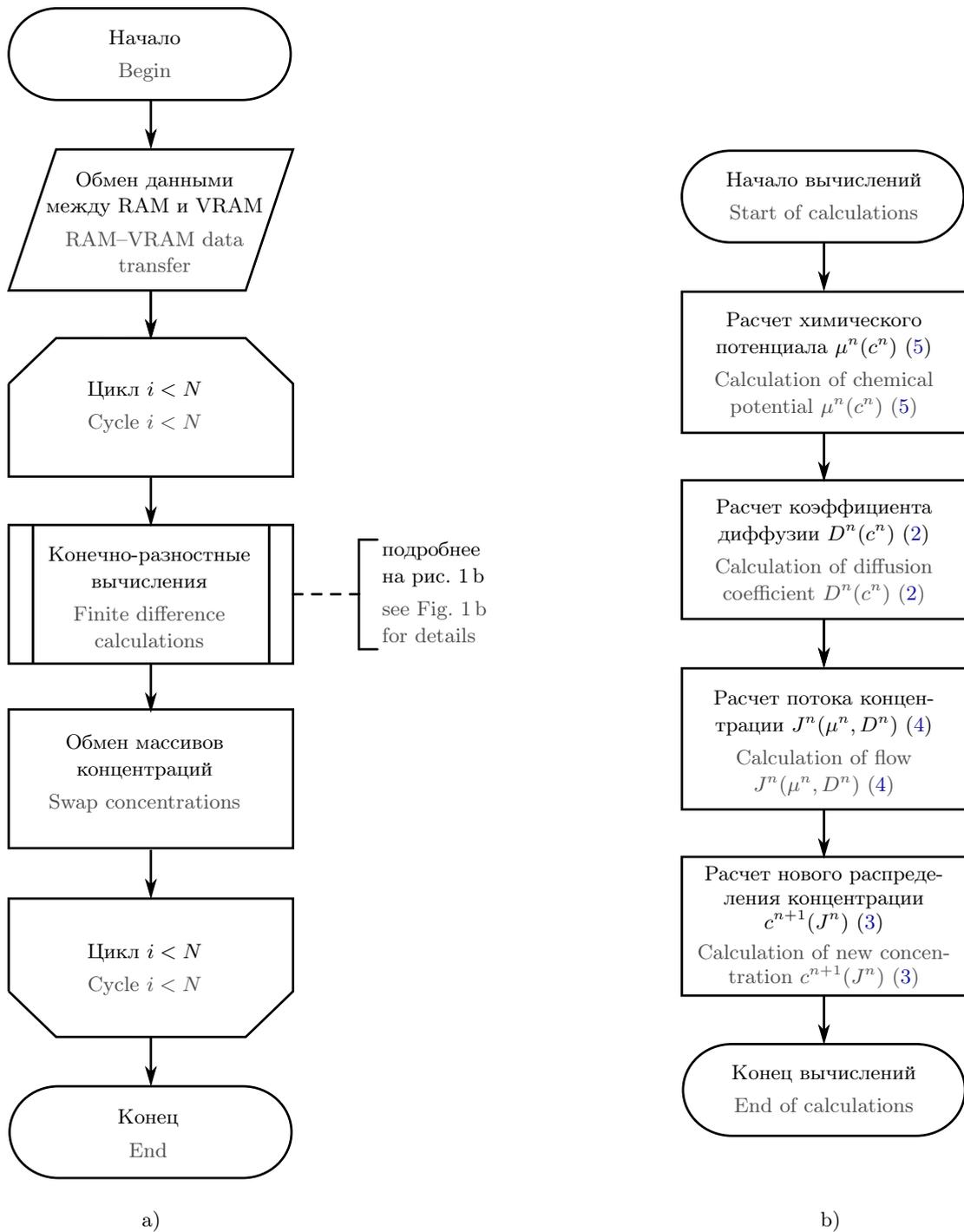


Рис. 1. Подпрограмма для решения уравнения Кана–Хиллиарда в отдельной подобласти на одном GPU: а) блок-схема всей подпрограммы; б) схема конечно-разностных вычислений

Fig. 1. Subroutine for solving the Cahn–Hilliard equation in a separate subdomain on a single GPU: а) flowchart of the entire subroutine; б) flowchart of finite difference calculations

Рассмотрим случай, когда c^{n+1} вычисляется только в некоторой прямоугольной подобласти (обозначена зеленым на рис. 2, для наглядности изображен двумерный случай) области Ω . Тогда поток J должен быть вычислен в подобласти, выделенной желтым, μ и D — в оранжевой, а исходное распределение c^n должно быть известно в области, обозначенной красным цветом. Поэтому для каждой подобласти необходимы два слоя дополнительных узлов.

4.2. Обмен данными между MPI-процессами. Для того чтобы уменьшить время работы алгоритма, подпрограмма конечно-разностных расчетов выполняется сначала на границах подобластей и затем, пока происходит обмен вычисленными значениями, новое значение c^{n+1} вычисляется во всех остальных внутренних узлах конечно-разностной сетки.

Рассмотрим подробную схему источников данных в различных узлах дополнительных слоев. На рис. 3 зеленые и желтые области соответствуют данным из соседних по ребру (по грани в трехмерном случае) подобластей. В узлы, выделенные оранжевым, значения попадают из подобластей, имеющих общую вершину (общее ребро в 3D) с данной подобластью. Следовательно, в трехмерном случае каждая подобласть обменивается данными с 18 другими подобластями (6 соседей по грани и 12 по ребру). Выполнение большого количества обменов в данном случае вычислительно неэффективно. Вместо этого используется MPI-коммуникатор с трехмерной периодической декартовой топологией, а обмен данными в дополнительных узлах проводится последовательно с 6 соседними по граням подобластями. Например, в двумерном случае (рис. 4) сначала данные из узлов, выделенных желтым, обновляются из соседних по горизонтальному направлению подобластей, затем данные в угловых узлах, выделенных оранжевым, перемещаются в вертикальном направлении вместе с данными из узлов, выделенных зеленым. MPI-обмены выполняются последовательно, после первого обмена данные, которые должны быть отправлены в диагональном направлении, копируются в буферы отправки для второго и третьего направления. Аналогично после второго обмена полученные данные из угловых узлов копируются в буфер обмена, соответствующий третьему направлению.

Чтобы отделить вычисления во внутренней части области от вычислений на границе, для каждого этапа конечно-разностных вычислений определяется область, в которой данный этап должен быть выполнен. На рис. 5 показано такое разделение вычислительной области для двумерного случая. На рис. 5а изображены области, в которых проводится вычисление химического потенциала μ и коэффициента диффузии D . В соответствии с разностной схемой вычислять значения в двух из четырех углов желтых областей не требуется. Тем не менее данные узлы включены в желтую область, так как CUDA-сетка имеет прямоугольную форму. На рис. 5б изображены области, в которых вычисляются компоненты потока J и обновленные значения концентрации c^{n+1} . Отметим, что компоненты потока J определены в полуцелых узлах, а концентрация c^{n+1} — в целых.

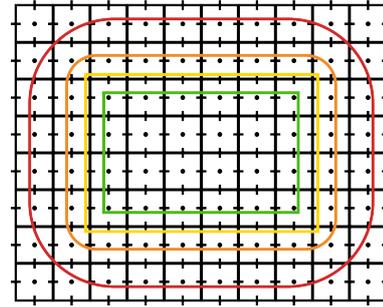
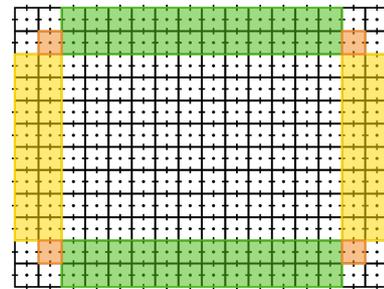
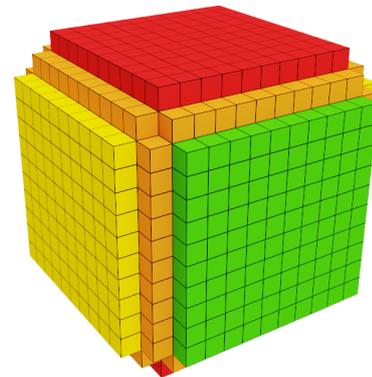


Рис. 2. Области вычисления параметров уравнения Кана–Хиллиарда на сдвинутых сетках: c^n — красная, μ , D — оранжевая, J — желтая, c^{n+1} — зеленая

Fig. 2. Computational areas of parameters on a staggered grid: c^n — red, μ , D — orange, J — yellow, c^{n+1} — green



a)



b)

Рис. 3. Схема получения данных для дополнительных слоев: а) двумерный случай; б) трехмерный случай. Зеленый, желтый и красный цвета соответствуют данным, получаемым из соседних по грани (ребру) подобластей, оранжевый — из соседних по вершине (ребру) подобластей

Fig. 3. Sources of halos: a) 2D; b) 3D. Green, yellow and red from neighbors by edge (face), orange from neighbors by vertex (edge)



В итоге новое распределение концентрации c^{n+1} вычисляется в желтых, зеленых и красной областях, причем ни в одном узле никакой из промежуточных параметров μ, D, J не вычисляется повторно. Вызовы `cudaMemcpy` и процедуры упаковки и распаковки также могут выполняться последовательно для каждого направления обмена. Поскольку обмен данными между RAM и VRAM занимает длительное время, возникает вопрос, имеет ли смысл минимизировать число вызовов `cudaMemcpy` или эффективнее выполнять обмен данными (как MPI-обмен, так и передачу данных VRAM-RAM) по заданному направлению во время вычисления нового значения концентрации для следующего направления?

Обе версии алгоритма используют два GPU-потока, которые выполняются независимо друг от друга, а также от основного потока процессора. MPI-обмены также выполняется асинхронно. Первая версия алгоритма (рис. 6) сначала вычисляет и упаковывает в GPU-буфер все данные для обмена. Затем происходит копирование данных в оперативную память, далее данные передаются необходимым MPI-процессам. После того как данные были получены, они копируются обратно в буфер графического процессора. Все это происходит во время вычислений во внутренней области. В конце итерации буфер распаковывается в массив вычислительной сетки. Вторая версия алгоритма (рис. 7) сначала вычисляет и упаковывает данные для первого обмена данными, а затем передает их во время вычислений для второго. Аналогично данные для третьего обмена вычисляются во время второго. В конце итерации все полученные данные перемещаются в буфер графического процессора за один вызов `cudaMemcpy`.

5. Оценка производительности. В данном разделе представлен анализ сильной и слабой масштабируемости разработанного алгоритма. Расчеты проводятся с использованием инфраструктуры “Политехник РСК Торнадо” (СПбГУ). Вычислительные узлы имеют два процессора Intel Xeon E5 2697 v3 и два графических ускорителя GPU NVIDIA Tesla K40 12 ГБ. Тестовой задачей является задача разделения фаз: в начальный момент времени в каждой точке пространства значение концентрации задается случайным образом (0 или 1, оба исхода равновероятны). Процесс, описываемый уравнением Кана–Хиллиарда, разделяет фазы (рис. 8). Безразмерные параметры модели принимают значения: $A = 16, \beta = 10, D_{vol} = 0.01, D_{var} = 0.001, D_{surf} = 16.0, \tau = 0.001, h_1 = h_2 = h_3 = 1$ [25]. Параметры числа потоков CUDA на один блок во всех экспериментах выбираются одинаковыми: $8 \times 8 \times 8$ во внутренней области, $16 \times 16 \times 2, 16 \times 2 \times 16$ и $2 \times 16 \times 16$ на соответствующих границах области.

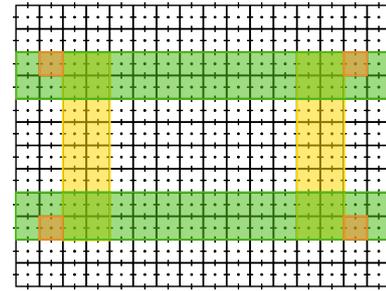
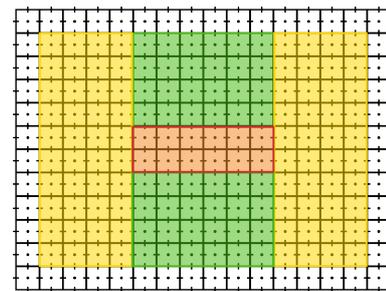
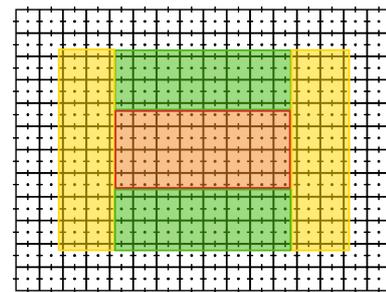


Рис. 4. Схема обмена данными в углах вычислительной сетки (оранжевый цвет) путем обменов дополнительными слоями узлов по двум направлениям

Fig. 4. The scheme of transferring corners (orange) through the two halo exchanges



a)



b)

Рис. 5. Области вычисления параметров: а) μ, D ; б) J, c^{n+1} . Желтый цвет — для обмена дополнительными слоями узлов по первому направлению, зеленый — по второму, красный — внутренняя область

Fig. 5. The areas of FD computation: а) μ, D ; б) J, c^{n+1} . Yellow is for the first halo exchange, green is for the second, and red is for the inner area

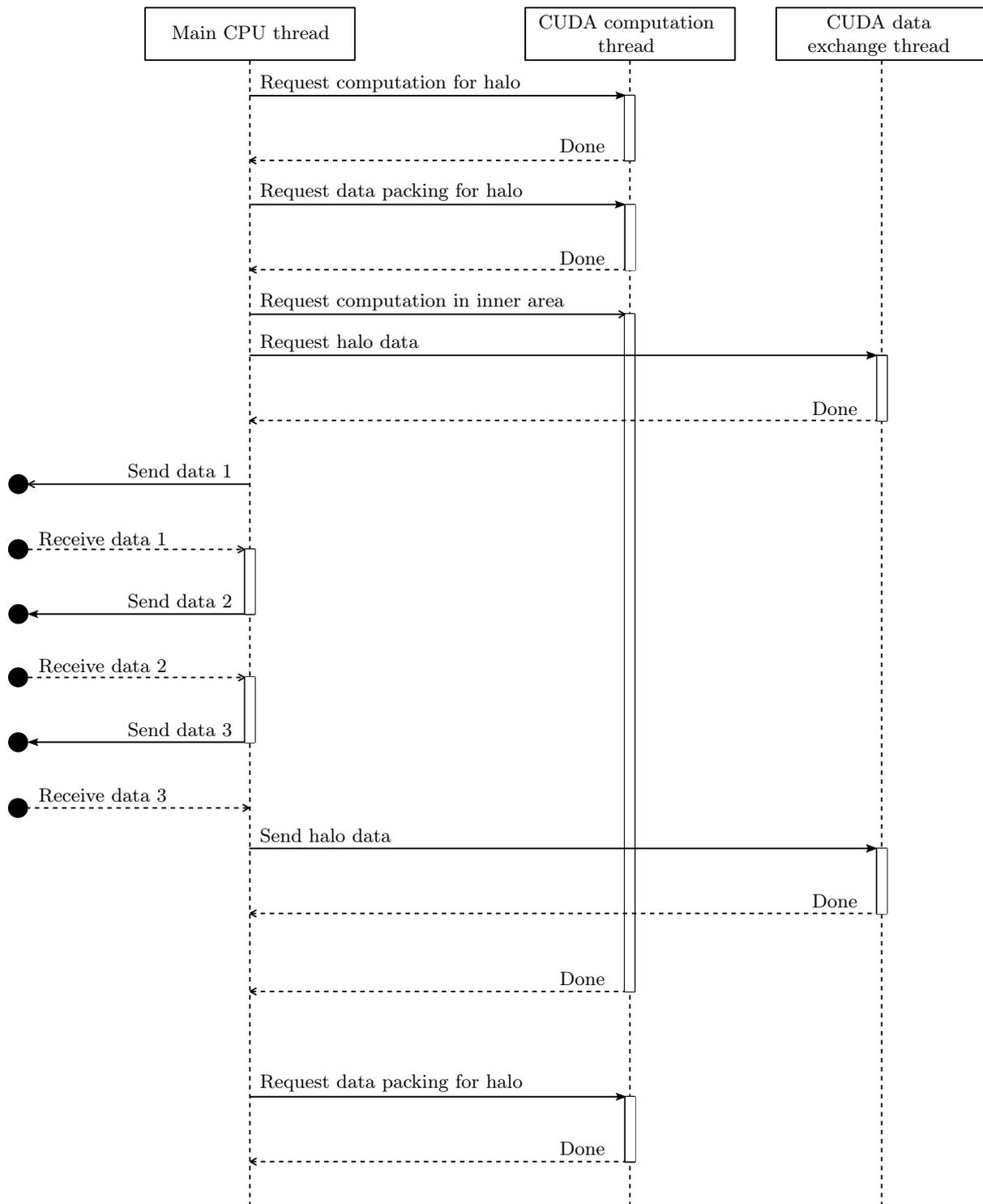


Рис. 6. Диаграмма последовательности для одного MPI-процесса первой версии алгоритма. Черными кругами обозначены MPI-процессы, с которыми обменивается данными текущий процесс. Все данные для обмена передаются из VRAM в RAM за один вызов `cudaMemcpy`

Fig. 6. The sequence diagram for the single MPI-process in the first version of the program. Black circles denote other MPI-processes, which communicate with current one. All halo data is transferred from device to host in a single `cudaMemcpy` call

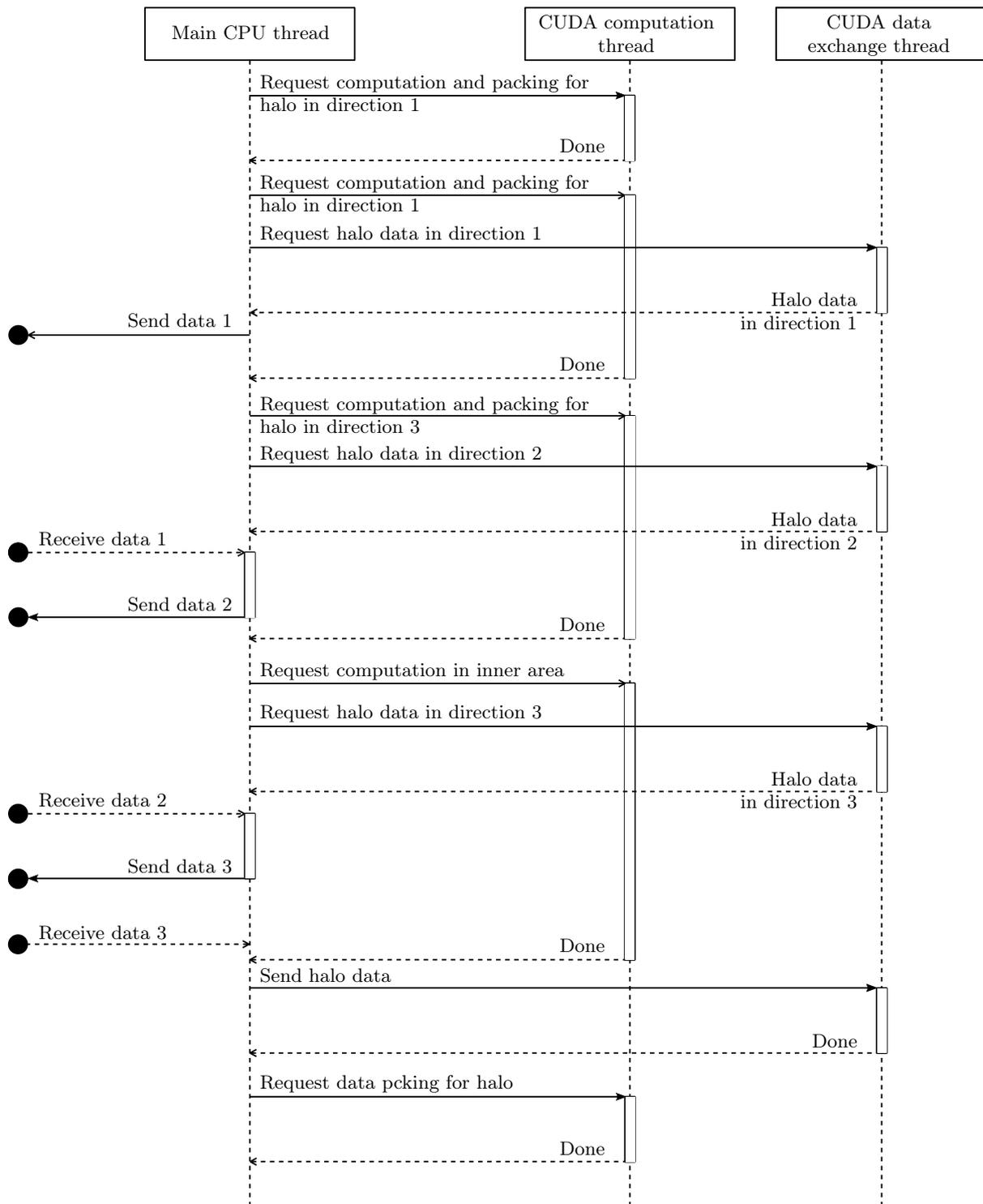


Рис. 7. Диаграмма последовательности для одного MPI-процесса второй версии алгоритма. Черными кругами обозначены MPI-процессы, с которыми обменивается данными текущий процесс. Все данные для обмена передаются из VRAM в RAM последовательно во время вычислений для передачи данных по другим направлениям

Fig. 7. The sequence diagram for the single MPI-process in the second version of the program. Black circles denote other MPI-processes, which communicate with current one. All halo data is transferred from device to host sequentially during the computation in other areas

5.1. Сильная масштабируемость. Для анализа сильной масштабируемости выбрана вычислительная сетка размером $516 \times 516 \times 516$ узлов. Следовательно, для хранения 7 параметров модели с использованием чисел двойной точности необходимо более 7 ГБ видеопамати. Помимо этого, для декомпозиции расчетной области необходимо хранить два дополнительных слоя узлов и соответствующие буферы обмена. Также для сохранения промежуточных состояний модели необходим еще один буфер, в который копируется концентрация из расчетной области без дополнительных узлов. Итого, общий размер массивов составляет $7 \times 520^3 + 516^3 + 4 \times (520^2 + 520 \times 516 + 516^2) = 1124864000$ элементов, или чуть менее чем 9 ГБ.

В первом эксперименте разделение фаз моделируется с использованием версии программы для одного графического ускорителя, т.е. для расчета периодических граничных условий напрямую берутся данные с противоположной границы области. Затем для каждой из описанных в разделе 4.2 схем обмена данными проводится аналогичный эксперимент с использованием 8, 12, 16, 27, 48 и 64 графических ускорителей и соответствующего числа MPI-процессов. График полученного ускорения представлен на рис. 9 а (Version 1 относится к схеме обмена данными, показанной на рис. 6, Version 2 — к схеме на рис. 7). Схема обмена, в которой сначала вычисляются данные для всех трех направлений и только затем происходит обмен между VRAM и RAM, показала лучшее ускорение и масштабируемость в сравнении со второй схемой. Как и ожидалось, с уменьшением размера подзадачи уменьшается и эффективность, поскольку растет отношение времени, занимаемого обменом данных, к времени вычислений.

Более информативной является оценка масштабируемости в сравнении с расчетом, выполненным на 8 графических ускорителях (рис. 9 б). В данной серии экспериментов оценивается производительность только первой “победившей” схемы обмена данными, а размер расчетной области увеличен до $1032 \times 1032 \times 1032$ узлов. Коэффициент ускорения для 27 MPI-процессов превышает $\frac{27}{8}$ (рис. 9 б). Это происходит из-за того, что при расчете подзадачи размером 516^3 блоки CUDA значительно выходят за пределы вычисли-

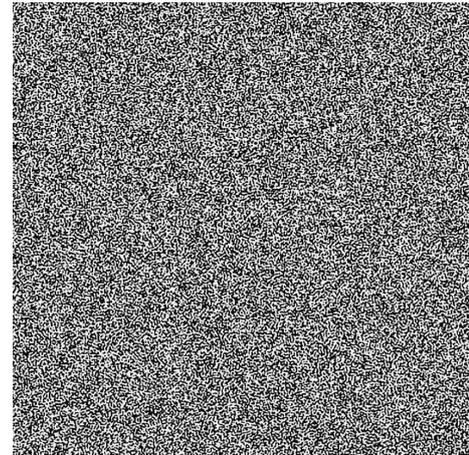


Рис. 8. Разделение фаз. Двумерный срез области размером 2064^3 узлов

Fig. 8. Phase separation. Slice of the 2064^3 domain

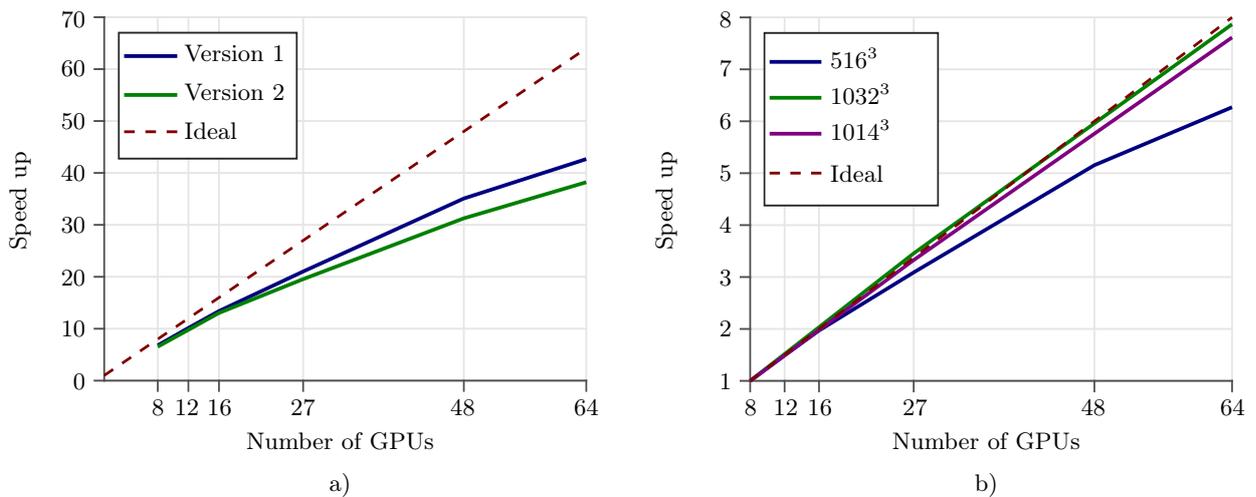


Рис. 9. Сильная масштабируемость алгоритма: а) цвет обозначает различные схемы обмена данными; б) цвет обозначает размер задачи

Fig. 9. Strong scaling of the algorithm: a) color denotes the kind of halo exchange; b) color denotes the size of the problem



тельной сетки, т.е. в эксперименте с 8 MPI-процессами значительная часть потоков CUDA, находящихся в вышедших за пределы вычислительной сетки блоках, не выполняет вычислений. Также это подтверждается оценками масштабируемости для задачи на вычислительной сетке размером $1014 \times 1014 \times 1014$ узлов, где коэффициент ускорения не превышает идеального случая линейной зависимости. Коэффициент эффективности для 64 графических ускорителей — 0.95.

5.2. Слабая масштабируемость. В случае гибридных вычислительных систем слабая масштабируемость алгоритма является более важным параметром, чем его сильная масштабируемость, поскольку использование графического процессора обеспечивает отличное распараллеливание и ускорение, однако накладывает существенные ограничения на объем используемой памяти, так как даже у современных графических процессоров (NVIDIA A100 80 ГБ, NVIDIA B200 192 ГБ) он существенно меньше, чем возможный объем RAM. Поэтому основная цель распараллеливания CUDA+MPI — это возможность решать большие задачи, объем которых превышает объем памяти одного графического ускорителя.

Для оценки слабой масштабируемости алгоритма с первой схемой обмена данными проведены эксперименты по численному моделированию разделения фаз с различными размерами вычислительной сетки (табл. 1). Во всех экспериментах одинаковы как размер подзадачи, так и соотношение сторон расчетной области для каждой из подзадач. Установлено, что увеличение размера задачи и изменение соотношения сторон расчетной области не влияют на производительность. Полученная эффективность почти линейна, а эффективность по отношению к версии программы для одного графического ускорителя составляет 0.86.

Отдельно стоит отметить, что подзадачи с кубическими расчетными областями — идеальный случай. Увеличение размера задачи при сохранении кубической формы расчетной области приводит к тому, что расчетные области для подзадач перестают быть кубическими (если число MPI-процессов не является кубом целого числа). Следовательно, увеличивается объем пересылаемых данных, что может привести к снижению коэффициента эффективности.

5.3. Производительность на графических ускорителях A100. Проведена серия расчетов на одном узле с 8 графическими ускорителями NVIDIA A100 (80 ГБ). Этого недостаточно для полноценного исследования масштабируемости, но это одна из конфигураций, в которой будет использоваться алгоритм. В первом расчете для решения задачи на вычислительной сетке размером 1000^3 используется один графический ускоритель без MPI-обменов. Во втором расчете для решения той же задачи используется

Таблица 1. Слабая масштабируемость

Table 1. Weak scaling

Число графических ускорителей GPU number	Размер вычислительной сетки Grid size	Время, с Time, s
8	$1032 \times 1032 \times 1032$	1906
12	$1548 \times 1032 \times 1032$	1907
16	$2064 \times 1032 \times 1032$	1903
27	$1548 \times 1548 \times 1548$	1911
48	$2064 \times 2064 \times 1548$	1938
64	$2064 \times 2064 \times 2064$	1915

Таблица 2. Результаты вычислений на графическом ускорителе A100

Table 2. A100 experiments

Число графических ускорителей GPU number	Размер вычислительной сетки Grid size	Время, с Time, s	Эффективность Efficiency
1	1000^3	2864	
8	1000^3	362	0.987
8	2000^3	4590	0.624

8 графических ускорителей. В третьем — число графических ускорителей остается прежним, а размер задачи увеличивается до 2000^3 .

Ускорение, полученное во втором расчете относительно первого, является почти идеальным (табл. 2). Для третьего расчета коэффициент эффективности достаточно низкий (0.62). Низкая эффективность может быть объяснена использованием одного узла. Узел оснащен двумя процессорами с 28 ядрами, что более чем достаточно для запуска MPI-процессов. Однако “бутылочным горлышком” для данной конфигурации может являться количество линий PCI Express.

6. Заключение. Для решения уравнения Кана–Хиллиарда разработан и реализован параллельный алгоритм, основанный на методе конечных разностей. Алгоритм ориентирован на использование гетерогенной вычислительной архитектуры. Исследованы два способа обмена данными между подзадачами. Тесты производительности показали, что предпочтительным способом обмена является перенос всех данных для дополнительных слоев сетки из видеопамяти в RAM с помощью одного вызова `cudaMemcpy`.

На примере задачи разделения фаз исследованы сильная и слабая масштабируемость алгоритма. Результаты экспериментов показывают хорошую масштабируемость алгоритма для подзадач с размером вычислительной сетки более 250^3 узлов. Однако производительность значительно снижается при дальнейшем уменьшении размера подзадачи. Следовательно, использование алгоритма для ускорения решения небольших задач не является целесообразным.

С другой стороны, изначальной целью разработки алгоритма было решение больших задач, требования к памяти которых значительно превосходят возможности отдельного графического ускорителя. Цель была достигнута, поскольку алгоритм демонстрирует почти идеальную слабую масштабируемость.

В дальнейшем параллельный алгоритм решения уравнения Кана–Хиллиарда будет использоваться вместе с решателем системы уравнений Навье–Стокса в программном комплексе для моделирования многофазных потоков в масштабе пор. Для этого будет добавлена возможность учитывать сложные граничные условия, возникающие в задачах цифровой физики горных пород. Еще одним направлением будущей работы является использование разработанного алгоритма для моделирования спекания.

Список литературы

1. Yang Y., Cai S., Yao J., et al. Pore-scale simulation of remaining oil distribution in 3D porous media affected by wettability and capillarity based on volume of fluid method // *Int. J. Multiph. Flow*. 2021. **143**, Article Number 103746. doi [10.1016/j.ijmultiphaseflow.2021.103746](https://doi.org/10.1016/j.ijmultiphaseflow.2021.103746).
2. Al-Khulaifi Y., Lin Q., Blunt M.J., Bijeljic B. Pore-scale dissolution by CO₂ saturated brine in a multimineral carbonate at reservoir conditions: impact of physical and chemical heterogeneity // *Water Resour. Res.* 2019. **55**, N 4. 3171–3193. doi [10.1029/2018wr024137](https://doi.org/10.1029/2018wr024137).
3. Prokhorov D., Lisitsa V., Khachkova T., et al. Topology-based characterization of chemically-induced pore space changes using reduction of 3D digital images // *J. Comput. Sci.* 2022. **58**, Article Number 101550. doi [10.1016/j.jocs.2021.101550](https://doi.org/10.1016/j.jocs.2021.101550).
4. Bazaikin Ya.V., Malkovich E.G., Prokhorov D.I., Derevschikov V.S. Detailed modeling of sorptive and textural properties of CaO-based sorbents with various porous structures // *Separation and Purification Technology*. 2021. **255**, Article Number 117746. doi [10.1016/j.seppur.2020.117746](https://doi.org/10.1016/j.seppur.2020.117746).
5. Hötzer J., Seiz M., Kellner M., et al. Phase-field simulation of solid state sintering // *Acta Materialia*. 2019. **164**. 184–195. doi [10.1016/j.actamat.2018.10.021](https://doi.org/10.1016/j.actamat.2018.10.021).
6. Mohammadi Alamooti A.H., Azizi Q., Davarzani H. Direct numerical simulation of trapped-phase recirculation at low capillary number // *Adv. Water Resour.* 2020. **145**, Article Number 103717. doi [10.1016/j.advwatres.2020.103717](https://doi.org/10.1016/j.advwatres.2020.103717).
7. Ji S., Tian C., Shi C., et al. New understanding on water–oil displacement efficiency in a high water-cut stage // *Petroleum Exploration and Development*. 2012. **39**, N 3. 362–370. doi [10.1016/S1876-3804\(12\)60052-4](https://doi.org/10.1016/S1876-3804(12)60052-4).
8. Kong D., Gao Y., Sarma H., et al. Experimental investigation of immiscible water-alternating-gas injection in ultra-high water-cut stage reservoir // *Advances in Geo-Energy Research*. 2021. **5**, N 2. 139–152. doi [10.46690/ager.2021.02.04](https://doi.org/10.46690/ager.2021.02.04).
9. Liu S.-Y., Ren B., Li H.-Y., et al. CO₂ storage with enhanced gas recovery (CSEGR): a review of experimental and numerical studies // *Petroleum Science*. 2022. **19**, N 2. 594–607. doi [10.1016/j.petsci.2021.12.009](https://doi.org/10.1016/j.petsci.2021.12.009).



10. Shen H., Yang Z., Li X., et al. CO₂-responsive agent for restraining gas channeling during CO₂ flooding in low permeability reservoirs // Fuel. 2021. **292**, Article Number 120306. doi 10.1016/j.fuel.2021.120306.
11. Derevschikov V.S., Prokhorov D.I., Bazaikin Ya.V., et al. Phenomenology and modeling of Y₂O₃ porous grain sintering // Ceramics International. 2023. **49**, N 6. 9452–9464. doi 10.1016/j.ceramint.2022.11.110.
12. Alizadeh M., Fatemi M. Mechanistic study of the effects of dynamic fluid/fluid and fluid/rock interactions during immiscible displacement of oil in porous media by low salinity water: direct numerical simulation // Journal of Molecular Liquids. 2021. **322**, Article Number 114544. doi 10.1016/j.molliq.2020.114544.
13. Balashov V., Zlotnik A. On a new spatial discretization for a regularized 3D compressible isothermal Navier–Stokes–Cahn–Hilliard system of equations with boundary conditions // J. Sci. Comput. 2021. **86**, N 3. Article Number 33. doi 10.1007/s10915-020-01388-6.
14. Yang Y.-F., Wang K., Lv Q.-F., et al. Flow simulation considering adsorption boundary layer based on digital rock and finite element method // Petroleum Science. 2021. **18**, N 1. 183–194. doi 10.1007/s12182-020-00476-4.
15. Pinilla A., Ramirez L., Asuaje M., Ratkovich N. Modelling of 3D viscous fingering: influence of the mesh on coreflood experiments // Fuel. 2021. **287**, Article Number 119441. doi 10.1016/j.fuel.2020.119441.
16. Zhu Q., Yang Y., Zhang X., et al. Pore-scale simulation of gas and water two-phase flow in rough-walled fractures using the volume of fluid method // Energies. 2022. **15**, N 24. Article Number 9382. doi 10.3390/en15249382.
17. Ferrari A., Magnini M., Thome J.R. A flexible coupled level set and volume of fluid (flexCLV) method to simulate microscale two-phase flow in non-uniform and unstructured meshes // Int. J. Multiphase Flow. 2017. **91**. 276–295. doi 10.1016/j.ijmultiphaseflow.2017.01.017.
18. Гадьяльщина К.А., Хачкова Т.С., Лисица В.В. Численное моделирование химического взаимодействия флюида с горной породой // Вычислительные методы и программирование. 2019. **20**, № 4. 457–470. doi 10.26089/NumMet.v20r440.
19. Zhang L., Xu C., Guo Y., et al. The effect of surface roughness on immiscible displacement using pore scale simulation // Transp. Porous Med. 2021. **140**, N 3. 713–725. doi 10.1007/s11242-020-01526-6.
20. Kim J. Phase-field models for multi-component fluid flows // Commun. Comput. Phys. 2012. **12**, N 3. 613–661. doi 10.4208/cicp.301110.040811a.
21. Tang M., Lu S., Zhan H., et al. The effect of a microscale fracture on dynamic capillary pressure of two-phase flow in porous media // Adv. Water Resour. 2018. **113**. 272–284. doi 10.1016/j.advwatres.2018.01.015.
22. Yang Q., Yao J., Huang Z., et al. Pore-scale investigation of petro-physical fluid behaviours based on multiphase SPH method // J. Pet. Sci. Eng. 2020. **192**. Article Number 107238. doi 10.1016/j.petrol.2020.107238.
23. Naik S., Gerke K.M., You Z., Bedrikovetsky P. Application of percolation, critical-path, and effective-medium theories for calculation of two-phase relative permeability // Physical Review E. 2021. **103**, N 4. Article Number 043306. doi 10.1103/PhysRevE.103.043306.
24. Badalassi V.E., Cenicerros H.D., Banerjee S. Computation of multiphase systems with phase field models // J. Comput. Phys. 2003. **190**, N 2. 371–397. doi 10.1016/S0021-9991(03)00280-8.
25. Wang Yu U. Computer modeling and simulation of solid-state sintering: a phase field approach // Acta Materialia. 2006. **54**, N 4. 953–961. doi 10.1016/j.actamat.2005.10.032.
26. Dong S. An efficient algorithm for incompressible N -phase flows // J. Comput. Phys. 2014. **276**. 691–728. doi 10.1016/j.jcp.2014.08.002.
27. Heida M., Málek J., Rajagopal K.R. On the development and generalizations of Cahn–Hilliard equations within a thermodynamic framework // Z. Angew. Math. Phys. 2012. **63**, N 1. 145–169. doi 10.1007/s00033-011-0139-y.
28. Dong S. Wall-bounded multiphase flows of N immiscible incompressible fluids: consistency and contact-angle boundary condition // J. Comput. Phys. 2017. **338**. 21–67. doi 10.1016/j.jcp.2017.02.048.
29. Sussman M., Smith K.M., Hussaini M.Y., et al. A sharp interface method for incompressible two-phase flows // J. Comput. Phys. 2007. **221**, N 2. 469–505. doi 10.1016/j.jcp.2006.06.020.
30. Molins S., Trebotich D., Yang L., et al. Pore-scale controls on calcite dissolution rates from flow-through laboratory and numerical experiments // Environ. Sci. Technol. 2014. **48**, N 13. 7453–7460. doi 10.1021/es5013438.
31. Cahn J.W., Hilliard J.E. Free energy of a nonuniform system. I. Interfacial free energy // J. Chem. Phys. 1958. **28**, N 2. 258–267. doi 10.1063/1.1744102.
32. Wang Z., Battiato I. A deep learning upscaling framework: reactive transport and mineral precipitation in fracture-matrix systems // Adv. Water Resour. 2024. **183**, Article Number 104588. doi 10.1016/j.advwatres.2023.104588.
33. Решетова Г.В., Хачкова Т.С. Численный метод оценки эффективных упругих характеристик горной породы по двумерным и трехмерным цифровым изображениям ядра // Вычислительные методы и программирование. 2017. **18**, № 4. 416–433. doi 10.26089/NumMet.v18r435.

34. *Crialesi-Esposito M., Scapin N., Demou A.D., et al.* FluTAS: a GPU-accelerated finite difference code for multiphase flows // *Comput. Phys. Commun.* 2023. **284**. Article Number 108602. doi [10.1016/j.cpc.2022.108602](https://doi.org/10.1016/j.cpc.2022.108602).
35. *Borgia R., Bestehorn M.* Phase-field model for Marangoni convection in liquid–gas systems with a deformable interface // *Phys. Rev. E*. 2003. **67**, N 6. Article Number 066307. doi [10.1103/PhysRevE.67.066307](https://doi.org/10.1103/PhysRevE.67.066307).
36. *Прохоров Д.И., Базайкин Я.В., Лисица В.В.* Эффективный алгоритм решения системы уравнений Аллена–Кана и Кана–Хиллиарда: моделирование процесса спекания // *Вычислительные методы и программирование*. 2022. **23**, № 2. 75–94. doi [10.26089/NumMet.v23r206](https://doi.org/10.26089/NumMet.v23r206).

Поступила в редакцию
25 ноября 2024 г.

Принята к публикации
7 декабря 2024 г.

Информация об авторе

Дмитрий Игоревич Прохоров — к.ф.-м.н., научн. сотр.; Институт нефтегазовой геологии и геофизики имени А. А. Трофимука СО РАН, просп. Коптюга, 3, 630090, Новосибирск, Российская Федерация.

References

1. Y. Yang, S. Cai, J. Yao, et al., “Pore-Scale Simulation of Remaining Oil Distribution in 3D Porous Media Affected by Wettability and Capillarity Based on Volume of Fluid Method,” *Int. J. Multiph. Flow* **143**, Article Number 103746 (2021). doi [10.1016/j.ijmultiphaseflow.2021.103746](https://doi.org/10.1016/j.ijmultiphaseflow.2021.103746).
2. Y. Al-Khulaifi, Q. Lin, M. J. Blunt, and B. Bijeljic, “Pore-Scale Dissolution by CO₂ Saturated Brine in a Multimineral Carbonate at Reservoir Conditions: Impact of Physical and Chemical Heterogeneity,” *Water Resour. Res.* **55** (4), 3171–3193 (2019). doi [10.1029/2018wr024137](https://doi.org/10.1029/2018wr024137).
3. D. Prokhorov, V. Lisitsa, T. Khachkova, et al., “Topology-Based Characterization of Chemically-Induced Pore Space Changes Using Reduction of 3D Digital Images,” *J. Comput. Sci.* **58**, Article Number 101550 (2022). doi [10.1016/j.jocs.2021.101550](https://doi.org/10.1016/j.jocs.2021.101550).
4. Ya. V. Bazaikin, E. G. Malkovich, D. I. Prokhorov, and V. S. Derevschikov, “Detailed Modeling of Sorptive and Textural Properties of CaO-Based Sorbents with Various Porous Structures,” *Sep. Purif. Technol.* **255**, Article Number 117746 (2021). doi [10.1016/j.seppur.2020.117746](https://doi.org/10.1016/j.seppur.2020.117746).
5. J. Hötzer, M. Seiz, M. Kellner, et al., “Phase-Field Simulation of Solid State Sintering,” *Acta Mater.* **164**, 184–195 (2019). doi [10.1016/j.actamat.2018.10.021](https://doi.org/10.1016/j.actamat.2018.10.021).
6. A. H. Mohammadi Alamooti, Q. Azizi, and H. Davarzani, “Direct Numerical Simulation of Trapped-Phase Recirculation at Low Capillary Number,” *Adv. Water Resour.* **145**, Article Number 103717 (2020). doi [10.1016/j.adwatre.2020.103717](https://doi.org/10.1016/j.adwatre.2020.103717).
7. S. Ji, C. Tian, C. Shi, et al., “New Understanding on Water–Oil Displacement Efficiency in a High Water-Cut Stage,” *Petrol. Explor. Develop.* **39** (3), 362–370 (2012). doi [10.1016/S1876-3804\(12\)60052-4](https://doi.org/10.1016/S1876-3804(12)60052-4).
8. D. Kong, Y. Gao, H. Sarma, et al., “Experimental Investigation of Immiscible Water-Alternating-Gas Injection in Ultra-High Water-Cut Stage Reservoir,” *Adv. Geo-Energy Res.* **5** (2), 139–152 (2021). doi [10.46690/ager.2021.02.04](https://doi.org/10.46690/ager.2021.02.04).
9. S.-Y. Liu, B. Ren, H.-Y. Li, et al., “CO₂ Storage with Enhanced Gas Recovery (CSEGR): A Review of Experimental and Numerical Studies,” *Pet. Sci.* **19** (2), 594–607 (2022). doi [10.1016/j.petsci.2021.12.009](https://doi.org/10.1016/j.petsci.2021.12.009).
10. H. Shen, Z. Yang, X. Li, et al., “CO₂-Responsive Agent for Restraining Gas Channeling during CO₂ Flooding in Low Permeability Reservoirs,” *Fuel* **292**, Article Number 120306 (2021). doi [10.1016/j.fuel.2021.120306](https://doi.org/10.1016/j.fuel.2021.120306).
11. V. S. Derevschikov, D. I. Prokhorov, Ya. V. Bazaikin, et al., “Phenomenology and Modeling of Y₂O₃ Porous Grain Sintering,” *Ceram. Int.* **49** (6), 9452–9464 (2023). doi [10.1016/j.ceramint.2022.11.110](https://doi.org/10.1016/j.ceramint.2022.11.110).
12. M. Alizadeh and M. Fatemi, “Mechanistic Study of the Effects of Dynamic Fluid/Fluid and Fluid/Rock Interactions during Immiscible Displacement of Oil in Porous Media by Low Salinity Water: Direct Numerical Simulation,” *J. Mol. Liq.* **322**, Article Number 114544 (2021). doi [10.1016/j.molliq.2020.114544](https://doi.org/10.1016/j.molliq.2020.114544).
13. V. Balashov and A. Zlotnik, “On a New Spatial Discretization for a Regularized 3D Compressible Isothermal Navier–Stokes–Cahn–Hilliard System of Equations with Boundary Conditions,” *J. Sci. Comput.* **86** (3), Article Number 33 (2021). doi [10.1007/s10915-020-01388-6](https://doi.org/10.1007/s10915-020-01388-6).



14. Y.-F. Yang, K. Wang, Q.-F. Lv, et al., “Flow Simulation Considering Adsorption Boundary Layer Based on Digital Rock and Finite Element Method,” *Pet. Sci.* **18** (1), 183–194 (2021). doi [10.1007/s12182-020-00476-4](https://doi.org/10.1007/s12182-020-00476-4).
15. A. Pinilla, L. Ramirez, M. Asuaje, and N. Ratkovich, “Modelling of 3D Viscous Fingering: Influence of the Mesh on Coreflood Experiments,” *Fuel* **287**, Article Number 119441 (2021). doi [10.1016/j.fuel.2020.119441](https://doi.org/10.1016/j.fuel.2020.119441).
16. Q. Zhu, Y. Yang, X. Zhang, et al., “Pore-Scale Simulation of Gas and Water Two-Phase Flow in Rough-Walled Fractures Using the Volume of Fluid Method,” *Energies* **15** (24), Article Number 9382 (2022). doi [10.3390/en15249382](https://doi.org/10.3390/en15249382).
17. A. Ferrari, M. Magnini, and J. R. Thome, “A Flexible Coupled Level Set and Volume of Fluid (flexCLV) Method to Simulate Microscale Two-Phase Flow in Non-Uniform and Unstructured Meshes,” *Int. J. Multiph. Flow* **91**, 276–295 (2017). doi [10.1016/j.ijmultiphaseflow.2017.01.017](https://doi.org/10.1016/j.ijmultiphaseflow.2017.01.017).
18. K. A. Gadyshina, T. S. Khachkova, and V. V. Lisitsa, “Numerical Modeling of Chemical Interaction between a Fluid and Rocks,” *Numerical Methods and Programming*. **20** (4), 457–470 (2019). doi [10.26089/NumMet.v20r440](https://doi.org/10.26089/NumMet.v20r440).
19. L. Zhang, C. Xu, Y. Guo, et al., “The Effect of Surface Roughness on Immiscible Displacement Using Pore Scale Simulation,” *Transp. Porous Med.* **140** (3), 713–725 (2021). doi [10.1007/s11242-020-01526-6](https://doi.org/10.1007/s11242-020-01526-6).
20. J. Kim, “Phase-Field Models for Multi-Component Fluid Flows,” *Commun. Comput. Phys.* **12** (3), 613–661 (2012). doi [10.4208/cicp.301110.040811a](https://doi.org/10.4208/cicp.301110.040811a).
21. M. Tang, S. Lu, H. Zhan, et al., “The Effect of a Microscale Fracture on Dynamic Capillary Pressure of Two-Phase Flow in Porous Media,” *Adv. Water Resour.* **113**, 272–284 (2018). doi [10.1016/j.advwatres.2018.01.015](https://doi.org/10.1016/j.advwatres.2018.01.015).
22. Q. Yang, J. Yao, Z. Huang, et al., “Pore-Scale Investigation of Petro-Physical Fluid Behaviours Based on Multiphase SPH Method,” *J. Pet. Sci. Eng.* **192**, Article Number 107238 (2020). doi [10.1016/j.petro1.2020.107238](https://doi.org/10.1016/j.petro1.2020.107238).
23. S. Naik, K. M. Gerke, Z. You, and P. Bedrikovetsky, “Application of Percolation, Critical-Path, and Effective-Medium Theories for Calculation of Two-Phase Relative Permeability,” *Phys. Rev. E* **103** (4), Article Number 043306 (2021). doi [10.1103/PhysRevE.103.043306](https://doi.org/10.1103/PhysRevE.103.043306).
24. V. E. Badalassi, H. D. Ceniceros, and S. Banerjee, “Computation of Multiphase Systems with Phase Field Models,” *J. Comput. Phys.* **190** (2), 371–397 (2003). doi [10.1016/S0021-9991\(03\)00280-8](https://doi.org/10.1016/S0021-9991(03)00280-8).
25. Yu U. Wang, “Computer Modeling and Simulation of Solid-State Sintering: A Phase Field Approach,” *Acta Mater.* **54** (4), 953–961 (2006). doi [10.1016/j.actamat.2005.10.032](https://doi.org/10.1016/j.actamat.2005.10.032).
26. S. Dong, “An Efficient Algorithm for Incompressible N -Phase Flows,” *J. Comput. Phys.* **276**, 691–728 (2014). doi [10.1016/j.jcp.2014.08.002](https://doi.org/10.1016/j.jcp.2014.08.002).
27. M. Heida, J. Málek, and K. R. Rajagopal, “On the Development and Generalizations of Cahn–Hilliard Equations within a Thermodynamic Framework,” *Z. Angew. Math. Phys.* **63** (1), 145–169 (2012). doi [10.1007/s00033-011-0139-y](https://doi.org/10.1007/s00033-011-0139-y).
28. S. Dong, “Wall-Bounded Multiphase Flows of N Immiscible Incompressible Fluids: Consistency and Contact-Angle Boundary Condition,” *J. Comput. Phys.* **338**, 21–67 (2017). doi [10.1016/j.jcp.2017.02.048](https://doi.org/10.1016/j.jcp.2017.02.048).
29. M. Sussman, K. M. Smith, M. Y. Hussaini, et al., “A Sharp Interface Method for Incompressible Two-Phase Flows,” *J. Comput. Phys.* **221** (2), 469–505 (2007). doi [10.1016/j.jcp.2006.06.020](https://doi.org/10.1016/j.jcp.2006.06.020).
30. S. Molins, D. Trebotich, L. Yang, et al., “Pore-Scale Controls on Calcite Dissolution Rates from Flow-through Laboratory and Numerical Experiments,” *Environ. Sci. Technol.* **48** (13), 7453–7460 (2014). doi [10.1021/es5013438](https://doi.org/10.1021/es5013438).
31. J. W. Cahn and J. E. Hilliard, “Free Energy of a Nonuniform System. I. Interfacial Free Energy,” *J. Chem. Phys.* **28** (2), 258–267 (1958). doi [10.1063/1.1744102](https://doi.org/10.1063/1.1744102).
32. Z. Wang and I. Battiato, “A Deep Learning Upscaling Framework: Reactive Transport and Mineral Precipitation in Fracture-Matrix Systems,” *Adv. Water Resour.* **183**, Article Number 104588 (2024). doi [10.1016/j.advwatres.2023.104588](https://doi.org/10.1016/j.advwatres.2023.104588).
33. G. V. Reshetova and T. S. Khachkova, “A Numerical Method to Estimate the Effective Elastic Moduli of Rocks from Two- and Three-Dimensional Digital Images of Rock Core Samples,” *Numerical Methods and Programming* **18** (4), 416–433 (2017). doi [10.26089/NumMet.v18r435](https://doi.org/10.26089/NumMet.v18r435).
34. M. Crialesi-Esposito, N. Scapin, A. D. Demou, et al., “FluTAS: A GPU-Accelerated Finite Difference Code for Multiphase Flows,” *Comput. Phys. Commun.* **284**, Article Number 108602 (2023). doi [10.1016/j.cpc.2022.108602](https://doi.org/10.1016/j.cpc.2022.108602).



35. R. Borcia and M. Bestehorn, “Phase-Field Model for Marangoni Convection in Liquid–Gas Systems with a Deformable Interface,” *Phys. Rev. E* **67** (6), Article Number 066307 (2003). doi [10.1103/PhysRevE.67.066307](https://doi.org/10.1103/PhysRevE.67.066307).
36. D. I. Prokhorov, Ya. V. Bazaikin, and V. V. Lisitsa, “Efficient Algorithm for Solving the System of Allen–Cahn and Cahn–Hilliard Equations: Modeling the Sintering Process,” *Numerical Methods and Programming* **23** (2), 75–94 (2022). doi [10.26089/NumMet.v23r206](https://doi.org/10.26089/NumMet.v23r206).

Received
November 25, 2024

Accepted for publication
December 7, 2024

Information about the author

Dmitry I. Prokhorov — Ph.D., Research Scientist; Trofimuk Institute of Petroleum Geology and Geophysics SB RAS, Koptug prospekt, 3, 630090, Novosibirsk, Russia.