



doi 10.26089/NumMet.v26r432

УДК 519.632.4

## Численный алгоритм оценки параметра пористости образцов горных пород

**А. А. Манаев**

Институт математики имени С. Л. Соболева СО РАН,  
Новосибирск, Российская Федерация  
ORCID: 0009-0003-2948-3683, e-mail: [manaeff.leha@gmail.com](mailto:manaeff.leha@gmail.com)

**Т. С. Хачкова**

Институт математики имени С. Л. Соболева СО РАН,  
Новосибирск, Российская Федерация  
ORCID: 0000-0002-1595-7142, e-mail: [hachtanya@yandex.ru](mailto:hachtanya@yandex.ru)

**В. В. Лисица**

Институт математики имени С. Л. Соболева СО РАН,  
Новосибирск, Российская Федерация  
ORCID: 0000-0003-3544-4878, e-mail: [lisitsavv@yandex.ru](mailto:lisitsavv@yandex.ru)

**Аннотация:** В работе представлен численный алгоритм оценки параметра пористости для цифровых образцов горных пород, построенных с использованием микротомографических изображений. В основе алгоритма лежит численное решение трехмерного уравнения Пуассона с быстро изменяющимися высококонтрастными коэффициентами. Возникающая при этом система линейных алгебраических уравнений оказывается плохо обусловленной, поэтому для ускорения сходимости требуется использование предобуславливателя. Для решения применяется метод сопряженных градиентов, а предобуславливатель строится как обратный оператор Лапласа, соответствующий однородной модели. В свою очередь, для обращения используется спектральное разложение двух трехдиагональных матриц, соответствующих аппроксимации одномерных производных. Полученная серия одномерных задач решается методом прогонки. Используемый предобуславливатель может быть эффективно применен как к исходной задаче с быстро изменяющимися высококонтрастными коэффициентами, так и к задаче, в которой решение вычисляется только в поровом пространстве. В этом случае данный метод обеспечивает такую же точность результатов, как и исходный подход для неоднородных моделей, однако сходится почти вдвое быстрее исходного. Реализация с использованием графических процессоров позволяет решать задачи размером до  $10^9$  неизвестных с помощью одного GPU.

**Ключевые слова:** уравнение Пуассона, метод сопряженных градиентов, спектральное разложение.

**Благодарности:** Исследование выполнено при финансовой поддержке Российского научного фонда, грант № 22-11-00004-П (В. В. Лисицей разработан алгоритм, А. А. Манаевым предложена его программная реализация), а также в рамках государственного задания ИМ СО РАН FWNF-2022-0015 (Т. С. Хачковой проведены численные эксперименты).

**Для цитирования:** Манаев А.А., Хачкова Т.С., Лисица В.В. Численный алгоритм оценки параметра пористости образцов горных пород // Вычислительные методы и программирование. 2025. 26, № 4. 479–501. doi 10.26089/NumMet.v26r432.



## Numerical algorithm for estimating the formation factor of digital rocks

**Aleksei A. Manaev**

Sobolev Institute of Mathematics SB RAS,  
Novosibirsk, Russia

ORCID: 0009-0003-2948-3683, e-mail: [manaev.1eha@gmail.com](mailto:manaev.1eha@gmail.com)

**Tatyana S. Khachkova**

Sobolev Institute of Mathematics SB RAS,  
Novosibirsk, Russia

ORCID: 0000-0002-1595-7142, e-mail: [hachtanya@yandex.ru](mailto:hachtanya@yandex.ru)

**Vadim V. Lisitsa**

Sobolev Institute of Mathematics SB RAS,  
Novosibirsk, Russia

ORCID: 0000-0003-3544-4878, e-mail: [lisitsavv@yandex.ru](mailto:lisitsavv@yandex.ru)

**Abstract:** The paper presents a numerical algorithm of estimating the formation factor for digital rock samples constructed using microtomographic images. The algorithm is based on the numerical solution of the three-dimensional Poisson equation with rapidly changing high-contrast coefficients. The arising system of linear algebraic equations turns out to be ill-conditioned, so to speed up the convergence it is necessary to use a preconditioner. The conjugate gradient method is applied to solve the system and the preconditioner is constructed as the inverse Laplace operator corresponding to the homogeneous model. In its turn, the spectral decomposition of two tridiagonal matrices appearing in the approximation of one-dimensional derivatives is used for inversion. The resulting series of one-dimensional problems is solved by the Thomas algorithm. The used preconditioner can be effectively applied both to the original problem with rapidly changing high-contrast coefficients and to the problem, in which the solution is calculated only in the pore space. In the latter case, the method provides the same accuracy of results as the original one for inhomogeneous models, but it converges almost twice as fast. Implementation using graphics processors allows solving problems up to  $10^9$  unknowns with a single GPU.

**Keywords:** Poisson equation, conjugate gradient method, spectral decomposition.

**Acknowledgements:** The study was carried out with the financial support of the Russian Science Foundation, grant No. 22-11-00004-П (V. V. Lisitsa developed the algorithm, A. A. Manaev implemented it), and within the framework of the state assignment of the Sobolev Institute of Mathematics SB RAS under contract No. FWNF-2022-0015 (T. S. Khachkova conducted numerical experiments).

**For citation:** A. A. Manaev, T. S. Khachkova and V. V. Lisitsa, “Numerical algorithm for estimating the formation factor of digital rocks,” *Numerical Methods and Programming*. **26** (4), 479–501 (2025). doi 10.26089/NumMet.v26r432.

---

**1. Введение.** Численный апскейлинг является мощным инструментом для исследования различных физических процессов в пористых средах. Этот подход основан на численном моделировании физического процесса в масштабе пор с последующим усреднением результатов для замены пористого материала однородной сплошной средой. В частности, абсолютная проницаемость оценивается путем решения стационарного уравнения Стокса в пористой среде с последующим усреднением потока по образцу и требованием выполнения закона Дарси [1–3]. Относительная проницаемость и капиллярное давление оцениваются с помощью решения уравнения двухфазного течения в масштабе пор [4, 5]. Эффективные упругие модули находятся путем решения стационарных уравнений упругости [6, 7], а удельное электрическое сопротивление (или удельная проводимость) оценивается с помощью решения уравнения Пуассона в масштабе

пор и последующего усреднения потенциала с требованием выполнения закона Ома [8, 9]. Необходимо отметить, что эффективный коэффициент диффузии химических частиц в законе Фика, а также эффективную теплопроводность [10, 11] тоже можно оценить с помощью уравнения Пуассона, определяющего стационарное распределение концентрации и тепловых полей.

На основе оценки удельного электрического сопротивления (УЭС) пористых материалов определяется параметр пористости (или форм-фактор) как отношение УЭС пористого материала, заполненного жидкостью, к удельному сопротивлению жидкости. Это простейший параметр, характеризующий сложность геометрии и топологии порового пространства [12], который также используется для аппроксимации динамической проницаемости в уравнении пороупругости Био [13]. Кроме того, по той же схеме, что и электропроводность, определяются эффективный коэффициент диффузии и теплопроводность. Таким образом, их можно оценить как соответствующие параметры в свободном пространстве, деленные на параметр пористости. Это делает параметр пористости одной из основных характеристик пористого материала. Необходимо отметить, что существуют точные метрики, характеризующие геометрию и топологию порового пространства, такие как функционалы Минковского [2, 14] и числа Бетти [15] соответственно. Однако именно параметр пористости может быть напрямую использован для оценки эффективных характеристик, соответствующих различным физическим процессам.

Как упоминалось выше, определение параметра пористости требует решения уравнения Пуассона в моделях, соответствующих пористым средам. При этом в масштабе пор существуют два возможных способа описания пористого материала.

Первый предполагает рассмотрение всего образца с порами, заполненными жидкостью с определенной проводимостью, и твердой матрицей (возможно, неоднородной) с другой заданной проводимостью. Эта постановка является классической. Существует множество подходов к дискретизации такой задачи, включая метод конечных разностей, конечных элементов и другие, а также к решению полученной системы линейных алгебраических уравнений (СЛАУ) [16]. Однако при работе с пористыми средами возникают модели с быстро изменяющимися (длина корреляции около 5 узлов сетки) и высококонтрастными (отличающимися на несколько порядков величин) коэффициентами. В результате рассматриваемая СЛАУ оказывается плохо обусловленной, поэтому для ускорения сходимости требуется использование предобуславливателя. Классические подходы к построению предобуславливателей могут быть неэффективны для таких моделей. В частности, использование алгебраических подходов, таких как метод Якоби или неполная LU факторизация с малоранговыми аппроксимациями [17–19], не гарантирует заметного ускорения. Другим вариантом являются многосеточные подходы [20, 21]. Однако их производительность ухудшается с ростом сложности модели. В нашем исследовании рассматриваются предобуславливатели, обратные к приближенному оператору. В частности, мы следуем работам [22–25], где приближенный оператор является исходным, но соответствует однородным или одномерным слоистым средам. Такой оператор легко инвертировать с помощью псевдоспектральных методов, где для диагонализации оператора и его инверсии используются преобразования Фурье или синусоидальные преобразования [26]. Необходимо отметить, что псевдоспектральные методы широко используются для моделирования течений жидкости, где уравнение Пуассона с постоянными коэффициентами решается при определении давления. FFT-подходы (от англ. Fast Fourier Transform, т.е. быстрое преобразование Фурье) легко реализуются для периодических граничных условий, но в случае других типов граничных условий алгоритм требует дополнительных шагов и выделения памяти [26]. Чтобы преодолеть эту трудность, мы предлагаем использовать прямое разложение по собственным значениям одномерного оператора. Действительно, конечно-разностное приближение трехмерного уравнения Пуассона в однородных средах (одномерных слоистых средах) можно представить в виде тензорного произведения трех одномерных операторов с трехдиагональными матрицами. Легко вычислить их спектральное разложение и сохранить набор собственных значений и собственных векторов. Поскольку эти матрицы соответствуют одномерным задачам, то их размер пренебрежимо мал по сравнению с размером исходной задачи. Таким образом, преобразование, применяемое каждый раз, когда предобуславливатель воздействует на вектор, представляет собой лишь серию небольших задач по умножению матрицы на вектор.

Вторым возможным способом рассмотрения образца и оценки параметра пористости в моделях с высоким контрастом проводимости является решение уравнения Пуассона только в пространстве пор с использованием граничных условий Неймана на границе раздела поры-матрица. В данной работе показано, что разработанный алгоритм решения уравнений Пуассона методом сопряженных градиентов, предобусловленный обратным однородным оператором Лапласа, подходит для этой постановки. С одной стороны, он обеспечивает такую же точность результатов, как и исходный алгоритм для неоднородных моделей. С другой стороны, он сходится почти вдвое быстрее исходного алгоритма.

## 2. Постановка задачи.

**2.1. Дифференциальная задача.** Основной целью исследования является решение уравнения Пуассона в моделях с быстро меняющимися резкоконтрастными коэффициентами, что характерно для вычислительной физики горных пород и других пористых материалов. Рассмотрим вычислительную область в виде прямоугольного параллелепипеда, дискретизированную регулярной ортогональной сеткой, состоящей из вокселей. Математически эта задача может быть сформулирована следующим образом. Уравнение Пуассона

$$\nabla \cdot (\sigma(\mathbf{x}) \nabla \varphi(\mathbf{x})) = f(\mathbf{x}) \quad (1)$$

рассматривается внутри области  $D = [0, X_1] \times [0, X_2] \times [0, X_3]$ , где  $\mathbf{x} = (x_1, x_2, x_3)$ . Переменная  $\varphi$  может обозначать как электрический потенциал, так и температуру или давление жидкости в зависимости от конкретного моделируемого физического процесса. Величина  $0 < \sigma_{\min} \leq \sigma(\mathbf{x}) \leq \sigma_{\max} < \infty$  может быть коэффициентом электропроводности, теплопроводности или проницаемости жидкости. В данном исследовании рассматриваются изотропные модели, где  $\sigma$  — скалярный параметр, но в случае анизотропных сред он может быть также положительно определенным симметричным тензором. Правая часть  $f(\mathbf{x})$  зависит от постановки задачи: для моделей электрических или тепловых полей она обычно считается тривиальной, а для моделей течений жидкости — представляет собой остаточную дивергенцию скорости.

В численных экспериментах физики горных пород типичная постановка предполагает, что образец помещается в изолирующие или непроницаемые порты, что математически соответствует граничным условиям Неймана:

$$\begin{aligned} \nabla \varphi \cdot \mathbf{n}|_{x_2=0} &= 0, & \nabla \varphi \cdot \mathbf{n}|_{x_3=0} &= 0, \\ \nabla \varphi \cdot \mathbf{n}|_{x_2=X_2} &= 0, & \nabla \varphi \cdot \mathbf{n}|_{x_3=X_3} &= 0. \end{aligned} \quad (2)$$

Оставшееся направление считается направлением основного потока. Граничные условия на входе  $x_1 = 0$  и выходе  $x_1 = X_1$  определяются постановкой задачи. Это могут быть либо условия Дирихле на обеих границах

$$\varphi|_{x_1=0} = \Phi_L, \quad \varphi|_{x_1=X_1} = \Phi_R, \quad (3)$$

либо комбинация условий Дирихле (на выходе) и Неймана (на входе)

$$\nabla \varphi \cdot \mathbf{n}|_{x_1=0} = 0, \quad \varphi|_{x_1=X_1} = \Phi_R. \quad (4)$$

Первый тип условий (3) соответствует задаваемой разнице давлений или потенциалов на противоположных сторонах образца. Второй тип (4) описывает ситуацию с фиксированным потоком на входе. Предлагаемый подход позволяет без существенных изменений учитывать оба типа условий, поэтому сосредоточимся только на случае граничных условий Дирихле (3).

В приложении к вычислительной физике горных пород решение уравнения (1) можно интерпретировать как распределение электрического потенциала, поля концентрации, теплового поля или давления жидкости внутри неоднородного образца горной породы. Используя это решение, можно определить поток (электрический ток):

$$J_1 = \int_0^{X_2} \int_0^{X_3} \sigma(\mathbf{x}) \frac{\partial \varphi}{\partial x_1} dx_2 dx_3. \quad (5)$$

Если рассматривать эффективную однородную модель с фиксированной удельной проводимостью  $\Sigma$ , то можно решить уравнение (1) с граничными условиями (2) и (3) аналитически, получив

$$\varphi(\mathbf{x}) = \Phi_L + \frac{\Phi_R - \Phi_L}{X_1} x_1,$$

и определить электрический ток:

$$\tilde{J}_1 = \Sigma \frac{\Phi_R - \Phi_L}{X_1} S_1,$$

где  $S_1$  — площадь поверхности поперечного сечения образца, ортогонального к направлению  $x_1$ .

Предполагая, что заданная разность потенциалов на двух противоположных сторонах образцов должна вызывать одинаковый электрический ток в исходной неоднородной и эффективной однородной

средах, то есть  $\tilde{J}_1 = J_1$ , можно восстановить удельную электропроводность эффективного материала:

$$\Sigma = J_1 \frac{X_1}{S_1} \frac{1}{\Phi_R - \Phi_L}. \quad (6)$$

Необходимо также напомнить о параметре пористости материала, который можно интерпретировать как меру сложности топологии порового пространства. Он определяется как отношение удельной проводимости флюида, заполняющего поры, к оцененной эффективной удельной проводимости, т.е.

$$F = \frac{\sigma_{\text{ф}}}{\Sigma} = \frac{\sigma_{\text{ф}} S_1 (\Phi_R - \Phi_L)}{J_1 X_1}.$$

Таким образом, чем меньше параметр пористости, тем ближе образец к однородному.

**2.2. Конечно-разностная аппроксимация.** Для аппроксимации уравнения (1) конечными разностями вводится сетка со следующими узлами:

$$\begin{aligned} (x_1)_{i_1} &= h_1 i_1, & i_1 &= 0, \dots, N_1 + 1, \\ (x_2)_{i_2} &= h_2(i_2 - 1/2), & i_2 &= 1, \dots, N_2, \\ (x_3)_{i_3} &= h_3(i_3 - 1/2), & i_3 &= 1, \dots, N_3, \end{aligned}$$

где  $h_1$ ,  $h_2$  и  $h_3$  — шаги сетки вдоль соответствующих пространственных направлений. Такая геометрия сетки упрощает аппроксимацию граничных условий Дирихле, заданных при  $x_1 = 0$  и  $x_1 = X_1$ , и граничных условий Неймана, заданных на остальных гранях расчетной области. Также вводится ячейка сетки

$$C_{i_1, i_2, i_3} = [x_1 - h_1/2, x_1 + h_1/2] \times [x_2 - h_2/2, x_2 + h_2/2] \times [x_3 - h_3/2, x_3 + h_3/2]$$

и предполагается, что коэффициент  $\sigma(\mathbf{x})$  является постоянным внутри каждой ячейки. Это условие, как правило, используется в вычислительной физике горных пород, когда модели строятся по микротомографическим изображениям зерна.

После этого можно построить консервативную конечно-разностную схему второго порядка точности для аппроксимации уравнения Пуассона:

$$\begin{aligned} & \frac{1}{h_1} \left( \tilde{\sigma}_{j_1+1/2, j_2, j_3} \frac{\varphi_{j_1+1, j_2, j_3} - \varphi_{j_1, j_2, j_3}}{h_1} - \tilde{\sigma}_{j_1-1/2, j_2, j_3} \frac{\varphi_{j_1, j_2, j_3} - \varphi_{j_1-1, j_2, j_3}}{h_1} \right) \\ & + \frac{1}{h_2} \left( \tilde{\sigma}_{j_1, j_2+1/2, j_3} \frac{\varphi_{j_1, j_2+1, j_3} - \varphi_{j_1, j_2, j_3}}{h_2} - \tilde{\sigma}_{j_1, j_2-1/2, j_3} \frac{\varphi_{j_1, j_2, j_3} - \varphi_{j_1, j_2-1, j_3}}{h_2} \right) \\ & + \frac{1}{h_3} \left( \tilde{\sigma}_{j_1, j_2, j_3+1/2} \frac{\varphi_{j_1, j_2, j_3+1} - \varphi_{j_1, j_2, j_3}}{h_3} - \tilde{\sigma}_{j_1, j_2, j_3-1/2} \frac{\varphi_{j_1, j_2, j_3} - \varphi_{j_1, j_2, j_3-1}}{h_3} \right) = f_{j_1, j_2, j_3}, \quad (7) \end{aligned}$$

где

$$j_1 = 2, \dots, N_1 - 1, \quad j_2 = 2, \dots, N_2 - 1, \quad j_3 = 2, \dots, N_3 - 1.$$

Уравнение (7) справедливо для внутренних точек расчетной области. Для полноты картины представим аппроксимацию уравнения Пуассона для значений индексов, равных единице и сопоставляемых переменным  $x_1$  и  $x_2$  соответственно:

$$\begin{aligned} & \frac{1}{h_1} \left( \tilde{\sigma}_{3/2, j_2, j_3} \frac{\varphi_{2, j_2, j_3} - \varphi_{1, j_2, j_3}}{h_1} - \tilde{\sigma}_{1/2, j_2, j_3} \frac{\varphi_{1, j_2, j_3}}{h_1} \right) \\ & + \frac{1}{h_2} \left( \tilde{\sigma}_{1, j_2+1/2, j_3} \frac{\varphi_{1, j_2+1, j_3} - \varphi_{1, j_2, j_3}}{h_2} - \tilde{\sigma}_{1, j_2-1/2, j_3} \frac{\varphi_{1, j_2, j_3} - \varphi_{1, j_2-1, j_3}}{h_2} \right) \\ & + \frac{1}{h_3} \left( \tilde{\sigma}_{1, j_2, j_3+1/2} \frac{\varphi_{1, j_2, j_3+1} - \varphi_{1, j_2, j_3}}{h_3} - \tilde{\sigma}_{1, j_2, j_3-1/2} \frac{\varphi_{1, j_2, j_3} - \varphi_{1, j_2, j_3-1}}{h_3} \right) \\ & = f_{1, j_2, j_3} - \tilde{\sigma}_{1/2, j_2, j_3} \Phi_L / h_1^2, \end{aligned}$$

$$j_2 = 2, \dots, N_2 - 1, \quad j_3 = 2, \dots, N_3 - 1;$$

$$\begin{aligned} & \frac{1}{h_1} \left( \tilde{\sigma}_{j_1+1/2,1,j_3} \frac{\varphi_{j_1+1,1,j_3} - \varphi_{j_1,1,j_3}}{h_1} - \tilde{\sigma}_{j_1-1/2,1,j_3} \frac{\varphi_{j_1,1,j_3} - \varphi_{j_1-1,1,j_3}}{h_1} \right) \\ & + \frac{1}{h_2} \left( \tilde{\sigma}_{j_1,3/2,j_3} \frac{\varphi_{j_1,2,j_3} - \varphi_{j_1,1,j_3}}{h_2} \right) \\ & + \frac{1}{h_3} \left( \tilde{\sigma}_{j_1,1,j_3+1/2} \frac{\varphi_{j_1,1,j_3+1} - \varphi_{j_1,1,j_3}}{h_3} - \tilde{\sigma}_{j_1,1,j_3-1/2} \frac{\varphi_{j_1,1,j_3} - \varphi_{j_1,1,j_3-1}}{h_3} \right) = f_{j_1,1,j_3}, \\ & j_1 = 2, \dots, N_1 - 1, \quad j_3 = 2, \dots, N_3 - 1. \end{aligned}$$

Аппроксимации вблизи других границ строятся аналогичным образом. Коэффициенты  $\tilde{\sigma}$  определяются согласно правилу:

$$\begin{aligned} \tilde{\sigma}_{j_1 \pm 1/2, j_2, j_3} &= 2 \left( \frac{1}{\sigma_{j_1 \pm 1, j_2, j_3}} + \frac{1}{\sigma_{j_1, j_2, j_3}} \right)^{-1}, \\ \tilde{\sigma}_{j_1, j_2 \pm 1/2, j_3} &= 2 \left( \frac{1}{\sigma_{j_1, j_2 \pm 1, j_3}} + \frac{1}{\sigma_{j_1, j_2, j_3}} \right)^{-1}, \\ \tilde{\sigma}_{j_1, j_2, j_3 \pm 1/2} &= 2 \left( \frac{1}{\sigma_{j_1, j_2, j_3 \pm 1}} + \frac{1}{\sigma_{j_1, j_2, j_3}} \right)^{-1}, \end{aligned} \quad (8)$$

где

$$j_1 = 1, \dots, N_1 - 1, \quad j_2 = 1, \dots, N_2 - 1, \quad j_3 = 1, \dots, N_3 - 1.$$

Такое усреднение сохраняет второй порядок сходимости в случае, когда границы раздела поры-скелет совпадают с гранями ячеек сетки [27].

В итоге получается СЛАУ:

$$A\psi = g, \quad (9)$$

где  $\psi$  — векторное представление сеточной функции  $\varphi_{j_1, j_2, j_3}$ , такое, что  $\psi_k = \varphi_{j_1, j_2, j_3}$ , где  $k = j_3 + (j_2 - 1)N_3 + (j_1 - 1)N_2N_3$ . В данной статье используется указанный порядок индексов, однако индексы могут быть переставлены. Необходимо отметить, что матрица  $A$  является разреженной, самосопряженной и строго положительно определенной.

**2.3. Метод сопряженных градиентов с предобуславливателем.** Для решения СЛАУ (9) мы используем метод сопряженных градиентов с предобуславливателем [16] (см. алгоритм 1).

---

Алгоритм 1. Метод сопряженных градиентов с предобуславливателем

Algorithm 1. Method of conjugate gradient with preconditioner

---

```

1:  $\mathbf{r}_0 = \mathbf{g} - A\psi_0$ ,  $\mathbf{p}_0 = B^{-1}\mathbf{r}_0$ ,  $\mathbf{q}_0 = \mathbf{p}_0$ ;
2: while  $\|\mathbf{r}\|/\|\mathbf{r}_0\| > \varepsilon$  do
3:    $\alpha_j = \frac{(\mathbf{q}_j, \mathbf{r}_j)}{(A\mathbf{p}_j, \mathbf{p}_j)}$ ;
4:    $\psi_{j+1} = \psi_j + \alpha_j \mathbf{p}_j$ ;
5:    $\mathbf{r}_{j+1} = \mathbf{r}_j - \alpha_j A\mathbf{p}_j$ ;
6:    $\mathbf{q}_{j+1} = B^{-1}\mathbf{r}_{j+1}$ ;
7:    $\beta_j = \frac{(\mathbf{q}_{j+1}, \mathbf{r}_{j+1})}{(\mathbf{q}_j, \mathbf{r}_j)}$ ;
8:    $\mathbf{p}_{j+1} = \mathbf{q}_{j+1} + \beta_j \mathbf{p}_j$ ;
9:    $j = j + 1$ ;
10: end while
```

---

В этих обозначениях  $\psi_0$  — это начальное приближение, а  $\varepsilon$  — допустимая невязка. Для выполнения вычислений необходимо хранить в памяти пять векторов  $\psi_j$ ,  $\mathbf{r}_j$ ,  $\mathbf{q}_j$ ,  $\mathbf{p}_j$  и  $A\mathbf{p}_j$ . Два наиболее трудоемких шага алгоритма — это вычисление действия исходной матрицы  $A$  и предобуславливателя  $B^{-1}$  на вектор, т.е.  $A\mathbf{p}_j$  и  $B^{-1}\mathbf{r}_{j+1}$ . В этой работе в качестве предобуславливателя  $B$  рассматривается конечно-разностная аппроксимация уравнения Пуассона для однородных сред. Этот оператор относительно легко обратить, используя спектральный метод, т.е. применяя быстрое преобразование Фурье по всем направлениям, решая серии независимых уравнений и применяя обратное FFT. Другой вариант — применить FFT только по двум пространственным направлениям, а затем решить серию одномерных задач с трехдиагональными матрицами [8, 22, 25]. Однако мы рассматриваем ограниченную область с фиксированными (непериодическими) граничными условиями. Это значит, что требуются различные типы преобразований.



В частности, синус-преобразование Фурье, если применяются граничные условия Неймана [26], и косинус-преобразование — для граничных условий Дирихле. Эффективный алгоритм построения этих двух преобразований на основе FFT применительно к решению однородного уравнения Пуассона обсуждается в [26]. Отметим, что описанный подход сложен в применении, если используется комбинация условий Дирихле и Неймана. Более того, он неприменим в случае слоистых сред. Таким образом, в данной работе мы предлагаем использовать спектральное разложение одномерных приближений оператора Пуассона вдоль направлений  $x_2$  и  $x_3$ , а затем решить оставшуюся серию независимых систем вдоль направления  $x_1$ .

**3. Спектральный предобуславливатель.** Идея предлагаемого подхода заключается в использовании в качестве предобуславливателя дискретизированного оператора Лапласа, соответствующего однородной среде, т.е.  $B = A_0$ . В этом случае решение задачи о действии предобуславливателя  $B^{-1}\mathbf{r} = \mathbf{q}$  может быть переформулировано как решение СЛАУ  $B\mathbf{q} = \mathbf{r}$ . Эта система решается на каждой итерации алгоритма 1. Для нахождения неизвестных рассматриваемой системы необходимо представить ее в виде

$$B\mathbf{q} = (B_1 + B_2 + B_3)\mathbf{q} = \mathbf{r},$$

где матрицы  $B_j$  соответствуют аппроксимации производной по пространственному направлению  $x_j$ . Отметим, что векторы представляют собой трехмерные сеточные функции с одномерной индексацией, построенные по формуле  $k = j_3 + (j_2 - 1)N_3 + (j_1 - 1)N_2N_3$ , причем направление  $x_3$  является “быстрым”, а  $x_1$  — самым “медленным”. Таким образом, матрицы  $B_j$  имеют следующий вид:

$$B_1 = \frac{\sigma}{h_1^2} \begin{pmatrix} 2I & & & & & \\ & \ddots & & & & \\ & & 2I & & & \\ -I & & & -I & & \\ & \ddots & & & \ddots & \\ & & -I & & 2I & \\ & & & & & -I & \\ & & & & & & \ddots & \\ & & & & & & & -I & 2I \\ & & & & & & & & & \ddots & \\ & & & & & & & & & & -I & 2I \end{pmatrix},$$

$$B_2 = \frac{\sigma}{h_2^2} \begin{pmatrix} I & -I & & & & & & 0 \\ -I & 2I & -I & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & -I & 2I & -I & & & \\ & & & -I & I & & & 0 \\ & & & & & \ddots & & \\ 0 & & & & & & I & -I \\ & & & & & & -I & 2I & -I \\ & & & & & & & \ddots & \ddots & \ddots \\ & & & & & & & -I & 2I & -I \\ & & & & & & & & -I & I \end{pmatrix},$$

где  $\sigma = \max \sigma(\mathbf{x})$ ,  $I$  — единичная матрица размера  $N_3$ , а

$$B_3 = \frac{\sigma}{h_3^2} \left( \begin{array}{ccc|ccc} T_3 & & 0 & & & 0 \\ & \ddots & & & & \ddots \\ 0 & & T_3 & & & 0 \\ \hline & & & \ddots & & \\ 0 & & & & T_3 & 0 \\ \hline & & & & 0 & T_3 \end{array} \right),$$

где  $T_3$  — трехдиагональная матрица размера  $N_3 \times N_3$ :

$$T_3 = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}.$$

Необходимо обратить внимание, что матрицы  $B_1$  и  $B_2$  можно преобразовать к блочно-диагональному виду с трехдиагональными матрицами  $T_1$  и  $T_2$  на главной диагонали. Эти преобразования представляют собой просто перестановки строк и столбцов матриц, а именно:

$$T_1 = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}, \quad T_2 = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}.$$

Размеры матриц  $T_2$  и  $T_3$  равны  $N_2 \times N_2$  и  $N_3 \times N_3$  соответственно. Если обозначить матрицу, составленную из собственных векторов матрицы  $T_k$ , через  $V_k$ , то ее компоненты будем представлять в следующем виде:

$$V_k = \begin{pmatrix} v_{1,1}^{(k)} & v_{1,2}^{(k)} & \cdots & v_{1,N_k-1}^{(k)} & v_{1,N_k}^{(k)} \\ v_{2,1}^{(k)} & v_{2,2}^{(k)} & \cdots & v_{2,N_k-1}^{(k)} & v_{2,N_k}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_{N_k-1,1}^{(k)} & v_{N_k-1,2}^{(k)} & \cdots & v_{N_k-1,N_k-1}^{(k)} & v_{N_k-1,N_k}^{(k)} \\ v_{N_k,1}^{(k)} & v_{N_k,2}^{(k)} & \cdots & v_{N_k,N_k-1}^{(k)} & v_{N_k,N_k}^{(k)} \end{pmatrix}.$$

Собственные значения матриц  $B_2$  и  $B_3$  совпадают с собственными значениями  $T_2$  и  $T_3$  соответственно. Матрицы, составленные из собственных векторов матриц  $B_2$  и  $B_3$ , равны:

$$U_2 = \left( \begin{array}{ccc|ccc} v_{1,1}^{(2)} I & \cdots & v_{1,N_2}^{(2)} I & & & \\ \vdots & \ddots & \vdots & & & \\ v_{N_2,1}^{(2)} I & \cdots & v_{N_2,N_2}^{(2)} I & & & \\ \hline & & & \ddots & & \\ \hline & & & & v_{1,1}^{(2)} I & \cdots & v_{1,N_2}^{(2)} I \\ & & & & \vdots & \ddots & \vdots \\ & & & & v_{N_2,1}^{(2)} I & \cdots & v_{N_2,N_2}^{(2)} I \end{array} \right),$$



$$U_3 = \left( \begin{array}{cc|cc|cc} V_3 & & 0 & & & \\ & \ddots & & & & \\ 0 & & V_3 & & & \\ \hline & & & \ddots & & \\ 0 & & & & V_3 & 0 \\ & \ddots & & & & \\ & & 0 & & 0 & V_3 \end{array} \right).$$

Идея представленного подхода заключается в том, что матрицы получаются как тензорное произведение одномерных аппроксимаций. Таким образом, спектральное разложение  $B_k = U_k \Lambda_k U_k^T$  (где  $\Lambda_k$  — диагональная матрица, составленная из собственных значений  $B_k$ ) любой из трех рассматриваемых матриц не меняет две другие. Это означает, что можно применить разложения для  $B_2$  и  $B_3$  и получить:

$$U_2^T U_3^T (B_1 + B_2 + B_3) U_3 U_2 (U_2^T U_3^T \mathbf{q}) = (B_1 + \Lambda_2 + \Lambda_3) \boldsymbol{\gamma} = U_3^T U_2^T \mathbf{r} = \mathbf{s},$$

где  $\boldsymbol{\gamma} = U_2^T U_3^T \mathbf{q}$ . Таким образом, решение исходной задачи эквивалентно решению следующей:

$$(B_1 + \Lambda_2 + \Lambda_3) \boldsymbol{\gamma} = \mathbf{s},$$

где  $\Lambda_2$  и  $\Lambda_3$  — диагональные матрицы, а  $B_1$  — блочно-диагональная матрица с трехдиагональными блоками. Это означает, что задача распадается на  $N_2 N_3$  независимых систем вида:

$$(T_1 + d_{j_2}^{(2)} I + d_{j_3}^{(3)} I) \boldsymbol{\gamma}^{j_2, j_3} = \mathbf{s}^{j_2, j_3},$$

где  $d_{j_k}^{(k)}$  — собственные значения  $T_k$ ,  $j_k = 1, \dots, N_k$ , а векторы  $\boldsymbol{\gamma}^{j_2, j_3}$  и  $\mathbf{s}^{j_2, j_3}$  — размерности  $N_1$ . Здесь верхние индексы  $j_2, j_3$  используются для обозначения пары собственных значений  $d_{j_2}$  и  $d_{j_3}$ , которым соответствует рассматриваемая одномерная задача.

Алгоритм 2 описывает действие предобуславливателя на вектор  $\mathbf{q}$ :

Алгоритм 2. Действие предобуславливателя

Algorithm 2. Action of the preconditioner

---

```

1:  $\mathbf{s} = U_3^T U_2^T \mathbf{r}$ ;
2: for  $j_2 = 1, \dots, N_2$ 
3:   for  $j_3 = 1, \dots, N_3$ 
4:     Solve  $(T_1 + d_{j_2}^{(2)} I + d_{j_3}^{(3)} I) \boldsymbol{\gamma}^{j_2, j_3} = \mathbf{s}^{j_2, j_3}$ ;
5:   end for
6: end for
7:  $\mathbf{q} = U_2 U_3 \boldsymbol{\gamma}$ ;
```

---

В данном исследовании предполагается, что предобуславливатель построен как обратный лапласиан, соответствующий однородной среде. Спектральное разложение матриц  $T_2$  и  $T_3$  может быть выполнено аналитически:

$$d_{j_k}^{(k)} = 2 - 2 \cos \left( \frac{\pi j_k}{N_k} \right), \quad (10)$$

$$v_{j_k, m_k}^{(k)} = \sqrt{\frac{2}{N_k}} \cos \left( \frac{\pi}{N_k} m_k (j_k + 1/2) \right), \quad (11)$$

где  $k = 2, 3$ ,  $N_k$  — общее число точек в  $k$ -м направлении,  $j_k$  — число точек (компонент собственного вектора), а  $m_k$  — число собственных векторов. Можно также использовать другие граничные условия, что приведет к незначительному изменению аналитических выражений для собственных значений и собственных векторов.

**4. Реализация на базе графического процессора GPU.** Реализация алгоритма ориентирована на архитектуру на базе графического процессора, как и в работе [24]. Все переменные хранятся в памяти графического процессора, чтобы избежать передачи данных, которая замедляет работу алгоритма. Копирование памяти с устройства на хост и с хоста на устройство используется только в самом начале и

в самом конце процедуры. Необходимо загрузить массив коэффициентов  $\sigma_{j_1, j_2, j_3}$  в графический процессор и выделить память для остальных массивов. Также необходимо получить решение  $\psi$ , скопировав его из памяти графического процессора в оперативную память. Следует заметить, что память необходимо выделить для шести векторов: решения  $\psi$ , правой части  $\mathbf{g}$ , вспомогательных векторов для сопряженных градиентов  $\mathbf{r}$ ,  $\mathbf{p}$ ,  $\mathbf{q}$ ,  $\mathbf{s} = A\mathbf{p}$ . Размерность каждого вектора равна  $N_1 N_2 N_3$ , поэтому для его хранения требуется  $8N_1 N_2 N_3$  байтов. Кроме того, нам необходимо предварительно вычислить собственные значения и собственные векторы матриц  $T_2$  и  $T_3$ . Это можно сделать либо численно, либо с помощью аналитических выражений (10) и (11). Для хранения спектральных разложений требуется  $8(N_3 + N_2)$  байтов для собственных значений и  $8(N_2^2 + N_3^2)$  байтов для собственных векторов. Очевидно, что этот объем памяти пренебрежимо мал по сравнению с объемом трехмерных массивов (модель, решение, правая часть, вспомогательные векторы).

Алгоритм сопряженных градиентов с предобуславливателем 1 содержит четыре типа матричных и векторных операций. Во-первых, нужно вычислить действие исходного оператора на вектор, т.е. вычислить вектор  $A\mathbf{p}_j$ . Данная операция реализуется напрямую с помощью формулы (7). Эта процедура свободна от зависимостей по данным, поскольку основана на шаблонных вычислениях (stencil computations). Таким образом, ее распараллеливание на многоядерной архитектуре не представляет сложностей. Во-вторых, нужно вычислить скалярные произведения. Это можно сделать с помощью библиотеки cuBLAS, в частности, с использованием функции DOT. В-третьих, вычисление линейных комбинаций векторов, которые также выполняются с помощью функции AXPY из библиотеки cuBLAS. В-четвертых, учет действия предобуславливателя в соответствии с алгоритмом 2. Нам нужно применить замену переменных  $\mathbf{s} = U_3^T U_2^T \mathbf{r}$ , что эквивалентно многократному вычислению произведения матриц. Фактически, вместо хранения матриц  $U_2$  и  $U_3$  в явном виде, в памяти хранятся только компактные матрицы  $V_2$  и  $V_3$ . В частности, вычисление  $U_2 \mathbf{r}$  требует умножения матрицы размера  $N_2 \times N_2$  на матрицу размера  $N_2 \times N_1 N_3$ . То же самое относится и к действию  $U_3$ . Этот функционал можно рассматривать как умножение матрицы на матрицу, что хорошо оптимизируется процедурой cuBLAS GEMM. После этого должна быть решена серия из  $N_2 N_3$  СЛАУ, соответствующих одномерной задаче. Это можно сделать методом прогонки. Следует подчеркнуть, что можно решать каждую систему параллельно. Это означает, что для каждой пары собственных значений  $d_{j_2}^{(2)}$ ,  $d_{j_3}^{(3)}$  соответствующая система решается одним ядром. Поскольку предполагается, что модель (для предобуславливателя) однородна, нам не нужно хранить исходную трехдиагональную матрицу. Однако метод прогонки требует хранения одного одномерного массива коэффициентов для каждой пары собственных значений. Таким образом, в общей сложности нам потребуется дополнительно  $8N_1 N_2 N_3$  байтов для хранения временных коэффициентов. После построения серии решений  $\gamma^{j_2, j_3}$  можно восстановить весь вектор  $\gamma$ , а затем применить обратное преобразование  $\mathbf{q} = U_2 U_3 \gamma$  с помощью функций из библиотеки cuBLAS.

**5. Модификация алгоритма для высоких контрастов.** Отдельно стоит рассмотреть случай высоких контрастов коэффициентов. Будем считать, что  $\max \sigma(\mathbf{x}) = 1$ . В противном случае задачу можно обезразмерить. При этом предположим, что  $\sigma \approx 1$  соответствует удельной проводимости жидкости, заполняющей поровое пространство, а удельная проводимость твердого скелета на несколько порядков меньше, т.е.  $\sigma_g \ll 1$ . Тогда исходную задачу можно заменить следующей:

$$\nabla \cdot (\sigma(\mathbf{x}) \nabla \varphi(\mathbf{x})) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega_p,$$

где  $\Omega_p$  представляет собой поровое пространство с граничными условиями Неймана на границе пор и зерен  $\Gamma_{gp}$ . Граничные условия на входе и выходе те же, что и в исходной формулировке.

Рассмотренную конечно-разностную схему (7) можно легко адаптировать для решения модифицированной задачи. В самом деле, формулы (8) можно видоизменить следующим образом:

$$\tilde{\sigma}_{j_1+1/2, j_2, j_3} = \begin{cases} 2 \left( \frac{1}{\sigma_{j_1+1, j_2, j_3}} + \frac{1}{\sigma_{j_1, j_2, j_3}} \right)^{-1}, & \sigma_{j_1, j_2, j_3} > \sigma_{\min} \wedge \sigma_{j_1+1, j_2, j_3} > \sigma_{\min}, \\ 0, & \sigma_{j_1, j_2, j_3} \leq \sigma_{\min} \vee \sigma_{j_1+1, j_2, j_3} \leq \sigma_{\min}, \end{cases} \quad (12)$$

что, по сути, соответствует нулевому граничному условию Неймана на границе пор.

Следует отметить, что такой же подход к усреднению широко используется при моделировании сейсмического волнового поля, когда рассматривается граница раздела жидкость–твердое тело [28]. Более



того, можно полагать, что в случае  $\sigma_{j_1, j_2, j_3} \leq \sigma_{\min}$  в соответствующей точке выполняется уравнение  $\varphi_{j_1, j_2, j_3} = f_{j_1, j_2, j_3}$ , а не уравнение (7). Формально дискретизированный оператор распадается на несколько независимых операторов, соответствующих каждой изолированной поре. Число независимых задач равно нулевому числу Бетти порового пространства [2]. Кроме того, необходимо предположить, что каждая пора соединена с входной или выходной гранью, т.е. нет полностью изолированных пор. В противном случае матрица становится сингулярной из-за использования граничных условий Неймана на всех границах такой изолированной поры.

Важно подчеркнуть, что использование предложенного предобуславливателя возможно также при ограничении вычислений областью порового пространства. Т.е. предобуславливатель действует во всей области определения, тогда как решение строится только внутри пространства пор. В этом случае необходимо проецировать результат действия предобуславливателя на поровое пространство, а также вычислять внутренние произведения и нормы векторов только в этом пространстве, т.е. в точках, где  $\sigma_{j_1, j_2, j_3} > \sigma_{\min}$ .

Кроме того, данное утверждение справедливо не только для оценки удельного электрического сопротивления образцов горных пород, но и для моделирования многофазных течений флюидов в поровом пространстве. В последнем случае для обновления давления на каждом временном шаге решается уравнение Пуассона с граничными условиями Неймана на границе раздела пор и зерен [4, 29].

## 6. Численные эксперименты.

**6.1. Анализ производительности.** Эксперименты по проверке алгоритма воспроизводят результаты, представленные в работе [24], где рассматривались многослойные модели. В связи с этим мы опускаем их детальное описание. Перейдем сразу к экспериментам по анализу производительности. Сначала повторялись эксперименты, описанные в [24], и изучались зависимости числа итераций и времени вычислений от размера образца. Рассматривались два различных способа изменения размера модели. Во-первых, варьировался размер в направлении  $x_1$  независимо от двух других размеров. Во-вторых, варьировались  $N_2 = N_3$  независимо от значения  $N_1$ . Все размеры задавались в соответствии со следующим набором значений:  $N_j \in [20, 40, 80, 160, 320]$ , т.е. рассматривались 25 комбинаций моделей с размером  $N_1$  и  $N_2 = N_3$ . Для построения модели использовался генератор случайных чисел. Сначала модель размером  $N_1 \times 20^2$  вокселей заполнялась однородными кубическими блоками со стороной 4 вокселя, в которых удельная проводимость определялась как  $\sigma = 10^{R([-3, 1])}$ , где  $R$  обозначает однородное случайное распределение в интервале  $[-3, 1]$ . Далее размер итогового образца увеличивался за счет периодического тиражирования исходной структуры. Были сгенерированы 10 статистических реализаций модели, на которых проводилось численное моделирование с использованием разработанного алгоритма. При этом фиксировалось число итераций, необходимое для сходимости алгоритма. На рис. 1 представлен график зависимости числа итераций от размера модели. Стоит обратить внимание, что ранее описанный алгоритм, основанный на предобуславливании FFT [24], демонстрирует значительный рост числа итераций с увеличением размера модели вдоль направления  $x_1$ . В отличие от него, число итераций представленного в этой работе алгоритма несколько уменьшается с увеличением размера модели  $N_1$ . В то же время наблюдается небольшое увеличение числа итераций при увеличении количества точек в направлениях  $x_2$  и  $x_3$ .

Затем изучалась зависимость времени, необходимого для решения уравнения Пуассона, от размера модели. Фиксировалось чистое время решения системы уравнений без учета затрат времени на ввод-вывод, предварительные вычисления, оценку эффективной проводимости и т.д. На рис. 2, 3 представлены графики зависимости времени вычислений при использовании двух типов графических процессоров: RTX 2080Ti и A100. Можно заметить, что время, затраченное на решение задачи графическим процессором A100, составляет одну десятую от времени, затраченного на RTX 2080Ti. Время моделирования на RTX 2080Ti представлено в этой работе для демонстрации согласованности с ранее полученными результатами [24]. Указанные измерения позволяют оценить производительность алгоритма (или время на одну итерацию). Соответствующие кривые изображены на рис. 4 и 5. Видно, что при увеличении размера модели только в направлении  $x_1$  время одной итерации растет линейно, что согласуется с ожиданиями, поскольку в этом направлении задача решается с помощью алгоритма Томаса (см. рис. 4а, 5а). Если же размер модели фиксирован в направлении  $x_1$ , а изменяется в направлениях  $x_2$  и  $x_3$  (см. рис. 4b, 5b), то время на итерацию растет как  $N_2^2 N_3^2$ , то есть пропорционально  $N^4$ , где  $N = N_2 = N_3$ .

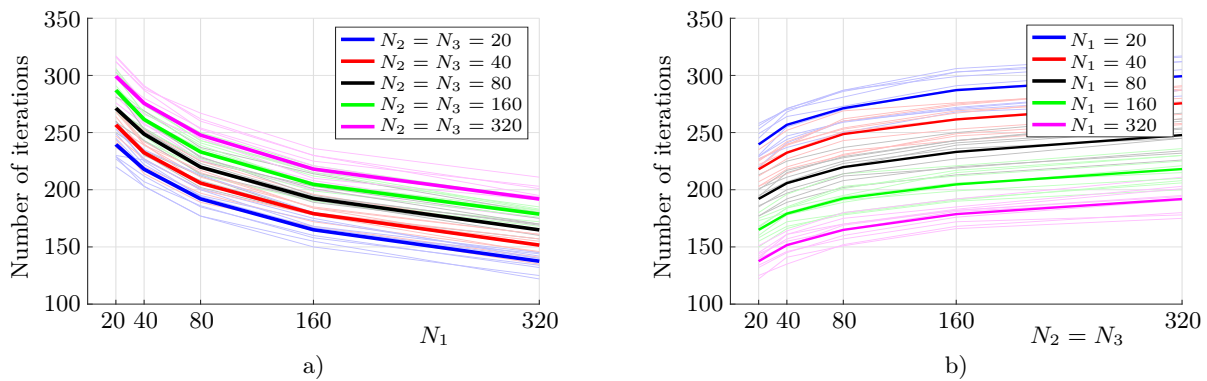


Рис. 1. Число итераций в зависимости от размера модели: а) варьируется  $N_1$  при фиксированных  $N_2 = N_3$ ; б) варьируются  $N_2 = N_3$  при фиксированном  $N_1$ . Толстые линии соответствуют среднему значению, тонкие линии — статистической реализации

Fig. 1. The number of iterations depending on the model size: а)  $N_1$  is varied at fixed  $N_2 = N_3$ ; б)  $N_2 = N_3$  are varied at fixed  $N_1$ . Thick lines represent mean values, thin lines correspond to a statistical realization

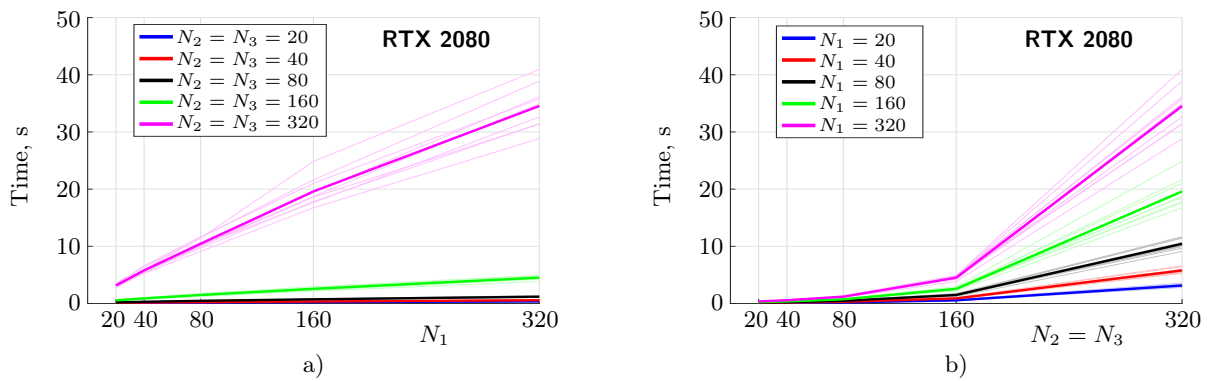


Рис. 2. Время решения уравнения Пуассона в зависимости от размера модели при использовании графического процессора RTX 2080 Ti: а) варьируется  $N_1$  при фиксированных  $N_2 = N_3$ ; б) варьируются  $N_2 = N_3$  при фиксированном  $N_1$ . Толстые линии соответствуют среднему значению, тонкие линии — статистической реализации

Fig. 2. Computational time for solving the Poisson equation on a GPU RTX 2080 Ti depending on the model size: а)  $N_1$  is varied at fixed  $N_2 = N_3$ ; б)  $N_2 = N_3$  are varied at fixed  $N_1$ . Thick lines represent mean values, thin lines correspond to a statistical realization

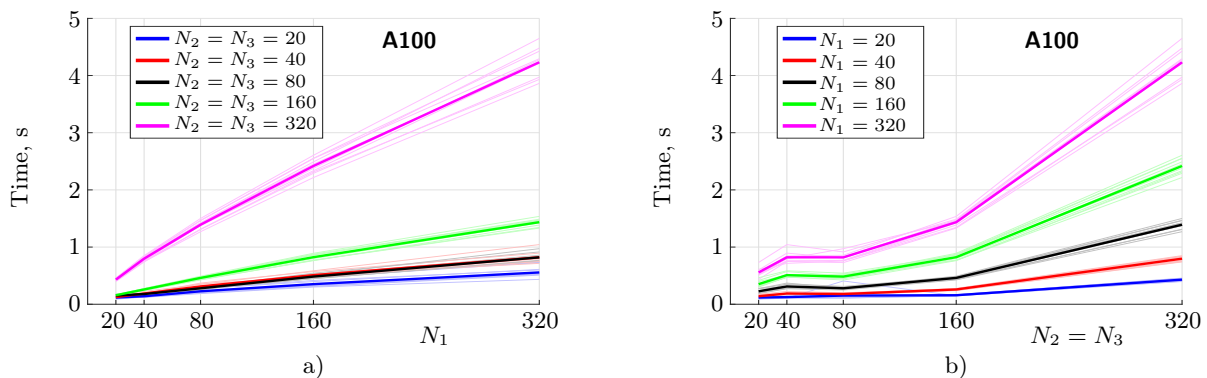


Рис. 3. Время решения уравнения Пуассона в зависимости от размера модели при использовании графического процессора A100: а) варьируется  $N_1$  при фиксированных  $N_2 = N_3$ ; б) варьируются  $N_2 = N_3$  при фиксированном  $N_1$ . Толстые линии соответствуют среднему значению, тонкие линии — статистической реализации

Fig. 3. Computational time for solving the Poisson equation on a GPU A100 depending on the model size: а)  $N_1$  is varied at fixed  $N_2 = N_3$ ; б)  $N_2 = N_3$  are varied at fixed  $N_1$ . Thick lines represent mean values, thin lines correspond to a statistical realization

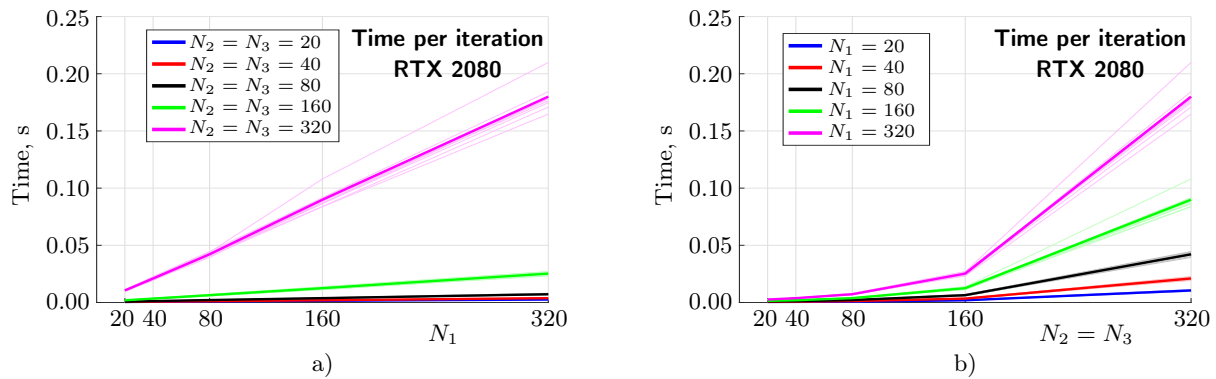


Рис. 4. Время на одну итерацию в зависимости от размера модели при использовании графического процессора RTX 2080 Ti: а) варьируется  $N_1$  при фиксированных  $N_2 = N_3$ ; б) варьируются  $N_2 = N_3$  при фиксированном  $N_1$ . Толстые линии соответствуют среднему значению, тонкие линии — статистической реализации

Fig. 4. Time per a single iteration in the case of GPU RTX 2080 Ti usage depending on the model size: а)  $N_1$  is varied at fixed  $N_2 = N_3$ ; б)  $N_2 = N_3$  are varied at fixed  $N_1$ . Thick lines represent mean values, thin lines correspond to a statistical realization

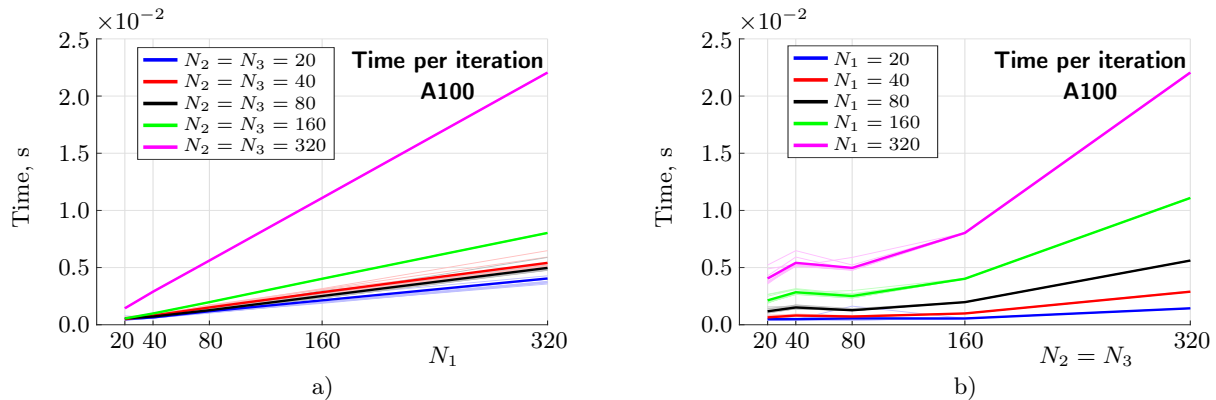


Рис. 5. Время на одну итерацию в зависимости от размера модели при использовании графического процессора A100: а) варьируется  $N_1$  при фиксированных  $N_2 = N_3$ ; б) варьируются  $N_2 = N_3$  при фиксированном  $N_1$ . Толстые линии соответствуют среднему значению, тонкие линии — статистической реализации

Fig. 5. Time per a single iteration in the case of GPU A100 usage depending on the model size: а)  $N_1$  is varied at fixed  $N_2 = N_3$ ; б)  $N_2 = N_3$  are varied at fixed  $N_1$ . Thick lines represent mean values, thin lines correspond to a statistical realization

**6.2. Влияние контраста коэффициентов.** Вторая серия экспериментов была проведена для исследования влияния контраста проводимости жидкости и скелета образца горной породы на скорость сходимости алгоритма. Мы воспроизвели эксперименты, представленные в [24], где сначала рассматривались две модели песчаника Бентхаймера (Bentheimer), подробно описанные в работе [2]. Первая (образец В) построена по КТ-изображениям с разрешением 5.58 мкм на воксель, разрешение второй (образец С) — 3.44 мкм на воксель. На рис. 6 (который воспроизводит рис. 4 из [24]) представлено поровое пространство цифровых образцов В и С. Исходные модели имели размер  $750^3$  вокселей. Однако, чтобы сделать эксперименты согласованными с ранее опубликованными, мы рассмотрели подобразцы размером  $400^3$  вокселей. Предполагалось, что поровое пространство насыщено жидкостью с удельной электропроводностью  $\sigma_d = 5$  См/м. При этом удельная проводимость матрицы  $\sigma_m$  варьировалась от  $10^{-7}$  до  $10^7$  См/м. На рис. 7 изображена зависимость числа итераций, необходимых для сходимости алгоритма, от проводимости матрицы, в табл. 1 и 2 представлено число итераций и время вычислений для двух типов графических процессоров GPU (RTX 2080Ti и A100) для образцов В и С соответственно. Для обоих образцов число итераций увеличивается с ростом контраста проводимости, поскольку число обусловленности матрицы увеличивается пропорционально контрасту. Однако, когда контраст становится очень большим, количе-

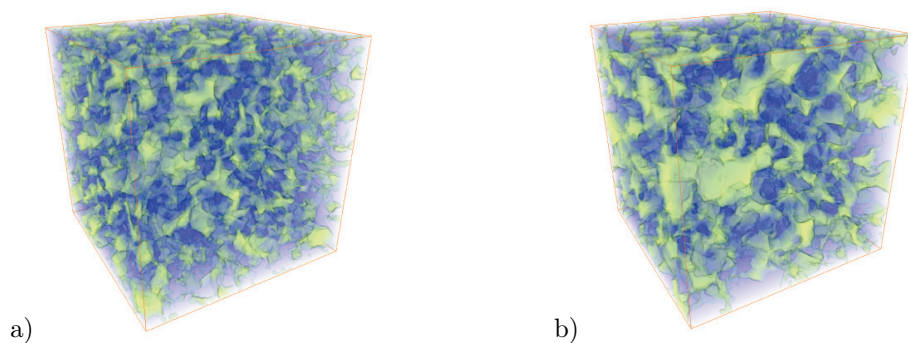


Рис. 6. Поровое пространство песчаника Бентхаймера: а) образец В с разрешением 5.58 мкм на воксель; б) образец С с разрешением 3.44 мкм на воксель

Fig. 6. Pore space of the Bentheimer sandstone: a) sample B with resolution 5.58  $\mu\text{m}$  per voxel; б) sample C with resolution 3.44  $\mu\text{m}$  per voxel

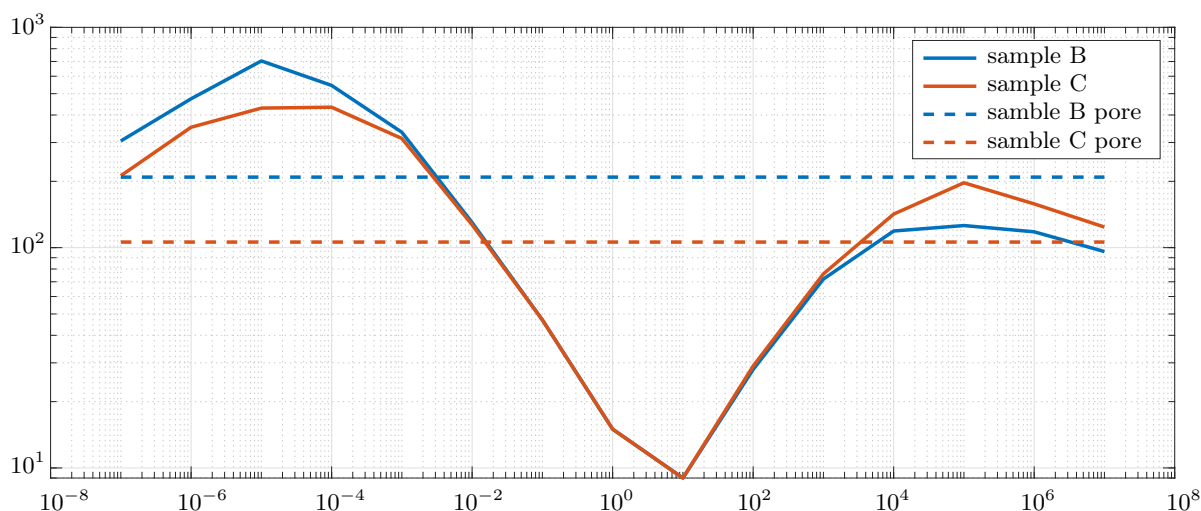


Рис. 7. Число итераций, необходимых для сходимости алгоритма, в зависимости от удельной проводимости матрицы  $\sigma_m$  (при постоянной  $\sigma_n = 5 \text{ См/м}$ )

Fig. 7. Number of iterations required for algorithm convergence depending on the matrix conductivity  $\sigma_m$  (at the fixed  $\sigma_n = 5 \text{ S/m}$ )

ство итераций уменьшается. Это можно объяснить уменьшением вклада точек с низкой проводимостью в общую невязку. Кроме того, это может означать снижение точности решения в этих точках. Но это решение практически не вносит вклада в оценку тока (5) и в оценку эффективной проводимости (6), что подтверждается оценками параметра пористости (FF), приведенными в табл. 1 и 2. Следует отметить, параметр пористости используется для оценки проводимости флюидозаполненного порового пространства, поэтому вычисляется в предположении, что проводимость матрицы много меньше проводимости флюида, заполняющего поровое пространство. Поэтому в табл. 1, 2 он вычислялся только для случаев  $\sigma_m \leq 10^{-2}$ .

Можно заметить, что параметр пористости практически не зависит от проводимости матрицы при  $\sigma_m < 10^{-3}$ . Это означает, что основной ток поддерживается жидкостью, заполняющей поры, а матрицу можно рассматривать как изолятор. Мы применили модифицированный алгоритм для решения уравнения Пуассона только внутри порового пространства, как описано в разделе 5. В этом случае число итераций, необходимых для сходимости алгоритма, составило 209 для образца В и 106 для образца С, что представлено горизонтальными пунктирными линиями на рис. 7. Следует подчеркнуть, что для обоих образцов модифицированный алгоритм сходился значительно быстрее исходного, особенно для “промежуточных” значений удельной проводимости матрицы  $\sigma_m \in [10^{-5}, 10^{-3}]$ . Полученные с помощью этого алгоритма значения параметра пористости составили 25.95 и 20.63 для образцов В и С соответственно.



Таблица 1. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом В

Table 1. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample B

$\sigma_m$	N it.	RTX 2080		A100		FF
		Time, s	Time/it., s	Time, s	Time/it., s	
$10^{-7}$	305	231.8	0.76	13.11	0.0430	25.95
$10^{-6}$	474	322.32	0.68	20.24	0.0427	25.96
$10^{-5}$	704	485.76	0.69	30.13	0.0428	25.94
$10^{-4}$	545	370.6	0.68	23.33	0.0428	25.82
$10^{-3}$	334	277.12	0.68	14.30	0.0428	24.71
$10^{-2}$	130	89.7	0.69	5.58	0.0429	18.61
$10^{-1}$	47	33.37	0.71	2.01	0.0428	
$10^0$	15	10.95	0.73	0.64	0.0427	
$10^1$	9	6.75	0.75	0.39	0.0432	
$10^2$	28	19.32	0.69	1.20	0.0428	
$10^3$	72	48.96	0.68	3.07	0.0427	
$10^4$	119	78.54	0.66	5.09	0.0428	
$10^5$	126	85.68	0.68	5.39	0.0428	
$10^6$	118	80.24	0.68	5.05	0.0428	
$10^7$	96	64.32	0.67	4.12	0.0429	

Таблица 2. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом С

Table 2. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample C

$\sigma_m$	N it.	RTX 2080		A100		FF
		Time, s	Time/it., s	Time, s	Time/it., s	
$10^{-7}$	212	138.64	0.87	9.07	0.0428	20.63
$10^{-6}$	352	231.47	0.66	15.03	0.0427	20.63
$10^{-5}$	430	283.28	0.66	18.40	0.0428	20.63
$10^{-4}$	434	299.7	0.69	18.58	0.0428	20.59
$10^{-3}$	313	212.48	0.68	13.40	0.0428	20.28
$10^{-2}$	127	86.63	0.68	5.45	0.0429	17.91
$10^{-1}$	47	32.02	0.68	2.02	0.0429	
$10^0$	15	10.83	0.72	0.65	0.0430	
$10^1$	9	7.42	0.87	0.39	0.0429	
$10^2$	29	20.23	0.7	1.24	0.0429	
$10^3$	76	50.76	0.67	3.25	0.0428	
$10^4$	142	94.25	0.66	6.08	0.0428	
$10^5$	197	129.26	0.66	8.43	0.0428	
$10^6$	158	106.15	0.67	6.76	0.0428	
$10^7$	124	84.54	0.68	5.31	0.0428	

Данный результат означает, что в случае экстремальных контрастов нет необходимости решать уравнение Пуассона внутри всей области. Чтобы получить соответствующую оценку, достаточно решить его только в поровом пространстве.

**6.3. Образцы со сложным поровым пространством.** Одним из наиболее сложных типов образцов для вычислительной физики горных пород является карбонатный, поскольку поры в таких образцах малы, и большая их часть при использовании современных микротомографов остается неразрешенной. Кроме того, разрешенные поры зачастую являются изолированными, а геометрия и топология связанных

пор обладают чрезвычайной сложностью. Чтобы проиллюстрировать применимость разработанного алгоритма к таким моделям, мы рассмотрели четыре набора цифровых образцов доломита из работы [30]. В процессе закачки  $\text{CO}_2$ , вызывающей химическое растворение породы, каждый образец экспериментаторы подвергали КТ-сканированию. В итоге для каждого из них получено по 10 изображений, а в общей сложности доступно всего 40 изображений. Необходимо отметить, что размер исходных цифровых образцов был около  $900^3$  вокселей. Чтобы провести численные эксперименты, подобные тем, что представлены в работе [24], из образцов вырезались подобразцы размером  $400^3$ . Поскольку они отличаются от подобразцов, рассмотренных в [24], это привело к небольшим расхождениям в оценке параметра пористости, однако основные тенденции сохранились. Следуя [30], мы обозначаем цифровые образцы аббревиатурами АН, АL, ВН, ВL — это образцы кальцита и доломита с неоднородностями А и В при высоких (Н) и низких (L) скоростях потока (более подробную информацию можно найти в [30]).

Для всех 40 образцов сначала оценивается открытая пористость, т.е. пренебрегаются изолированные поры, но учитываются поры, образующие связанное поровое пространство и поддерживающие перколяцию. Пористость каждого образца представлена в табл. 3–6. Для дальнейших численных экспериментов строятся расчетные модели со значениями удельной проводимости жидкости  $\sigma_n = 1 \text{ См/м}$  и матрицы  $\sigma_m = 10^{-4}$ . Необходимо заметить, что эти величины не являются фактическими значениями удельной проводимости

Таблица 3. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом АН

Table 3. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample АН

№ No.	por., %	FF	Исходный алгоритм Original algorithm			Модифицированный алгоритм Modified algorithm		
			N it.	Time, s	Time/it., s	N it.	Time, s	Time/it., s
1	4.41	282	477	20.37	0.0427	346	13.61	0.0393
2	5.70	157	484	20.69	0.0427	288	11.46	0.0398
3	9.36	44	477	20.31	0.0426	246	10.13	0.0412
4	12.21	23	473	20.07	0.0424	203	8.54	0.0421
5	13.92	17	478	20.34	0.0426	215	9.15	0.0426
6	14.83	15	478	20.32	0.0425	188	8.07	0.0429
7	15.76	13	480	20.36	0.0424	193	8.32	0.0431
8	15.80	13	480	20.42	0.0425	184	7.87	0.0428
9	16.26	12	483	20.72	0.0429	183	7.87	0.0430
10	16.94	11	484	20.79	0.0430	182	7.88	0.0433

Таблица 4. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом АL

Table 4. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample АL

№ No.	por., %	FF	Исходный алгоритм Original algorithm			Модифицированный алгоритм Modified algorithm		
			N it.	Time, s	Time/it., s	N it.	Time, s	Time/it., s
1	1.68	248	479	20.56	0.0429	521	20.25	0.0389
2	4.30	219	480	20.45	0.0426	509	20.13	0.0395
3	4.35	195	479	20.35	0.0425	564	22.14	0.0393
4	4.77	157	488	20.91	0.0428	423	16.71	0.0395
5	8.43	86	478	20.34	0.0426	398	16.13	0.0405
6	11.15	37	479	20.56	0.0429	424	17.57	0.0414
7	13.13	27	475	20.57	0.0433	293	12.35	0.0422
8	14.20	22	478	20.63	0.0432	303	12.9	0.0426
9	15.05	18	475	20.54	0.0432	286	12.33	0.0431
10	15.96	16	476	20.36	0.0428	278	12.05	0.0433



Таблица 5. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом ВН

Table 5. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample BH

№ No.	por., %	FF	Исходный алгоритм Original algorithm			Модифицированный алгоритм Modified algorithm		
			N it.	Time, s	Time/it., s	N it.	Time, s	Time/it., s
1	0.0	1201	497	21.25	0.0428	-	-	-
2	5.62	239	488	21.11	0.0433	354	14.06	0.0397
3	16.01	20.9	468	19.97	0.0427	211	9.10	0.0431
4	19.25	13.6	460	19.84	0.0431	201	8.88	0.0442
5	21.38	10.8	471	20.53	0.0436	171	7.68	0.0449
6	23.13	9.16	467	20.25	0.0434	202	9.22	0.0456
7	24.65	8.03	475	20.64	0.0435	191	8.75	0.0458
8	25.71	7.37	466	20.01	0.0429	151	7.05	0.0467
9	27.12	6.75	470	20.13	0.0428	164	7.70	0.0470
10	28.20	6.3	471	20.15	0.0428	180	8.49	0.0472

Таблица 6. Число итераций, время работы алгоритма, время на одну итерацию и параметр пористости для экспериментов с образцом BL

Table 6. Number of iterations, computational time, time per an iteration, and formation factor for the experiments with sample BL

№ No.	por., %	FF	Исходный алгоритм Original algorithm			Модифицированный алгоритм Modified algorithm		
			N it.	Time, s	Time/it., s	N it.	Time, s	Time/it., s
1	0.0	1175	490	21.36	0.436	-	-	-
2	1.31	1075	495	21.55	0.435	439	17.09	0.0389
3	1.42	921	494	21.72	0.440	379	14.75	0.0389
4	1.69	789	497	21.67	0.436	414	16.06	0.0388
5	1.75	768	498	21.81	0.438	423	16.36	0.0387
6	1.79	750	499	21.50	0.431	423	16.49	0.0390
7	1.83	725	500	21.44	0.429	416	16.19	0.0389
8	2.85	653	493	21.15	0.429	503	19.64	0.0390
9	2.90	645	495	21.71	0.439	507	19.79	0.0390
10	2.94	635	495	22.17	0.448	491	19.19	0.0391

доломита и кальцита; нас в данном случае интересует сложная геометрия порового пространства образцов. Согласно предыдущей серии экспериментов, для образцов с таким высоким контрастом проводимости можно использовать как исходный алгоритм, так и модифицированный, где решение строится только в поровом пространстве. Мы проводим моделирование с использованием обеих версий алгоритма, представляя результаты для четырех образцов в табл. 3–6. Оценка параметра пористости, полученная при помощи обеих версий алгоритма, совпадает для всех численных экспериментов с высокой точностью. Исключениями являются образцы ВН и ВЛ с нулевой открытой пористостью на первом этапе закачки. Для них моделирование по модифицированной версии не проводилось, а оценка, полученная по исходной версии, хотя и представлена, но не может считаться достоверной.

Число итераций и время на итерацию исходного алгоритма не зависят от пористости образца или форм фактора порового пространства (параметра пористости). В самом деле, как следует из предыдущих разделов, сходимость исходного алгоритма в первую очередь зависит от контрастности и лишь в меньшей степени — от размера образца. Напротив, модифицированный алгоритм сходится быстрее с увеличением пористости и уменьшением форм-фактора; таким образом, он чувствителен к геометрии и топологии порового пространства. Также следует отметить, что с ростом пористости увеличивается время выполнения одной итерации модифицированного алгоритма. На графиках зависимости числа итераций от открытой

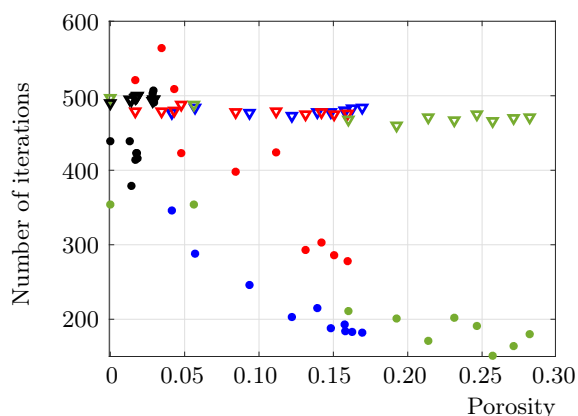


Рис. 8. Число итераций для 40 образцов карбонатов, рассчитанное исходным (треугольники) и модифицированным (точки) алгоритмами. Синие маркеры соответствуют образцам АН, красные — АЛ, зеленые — ВН, черные — ВЛ

Fig. 8. Number of iterations for 40 samples of carbonates, computed by the original (triangles) and modified (dots) algorithms. Blue markers correspond to АН samples, red markers — to АЛ, green — to ВН, and black — to ВЛ samples

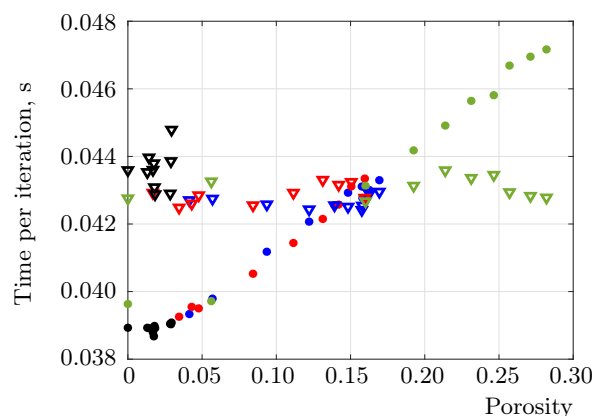


Рис. 9. Время на одну итерацию для 40 образцов карбонатов, рассчитанное исходным (треугольники) и модифицированным (точки) алгоритмами. Синие маркеры соответствуют образцам АН, красные — АЛ, зеленые — ВН, черные — ВЛ

Fig. 9. Time per a single iteration for 40 samples of carbonates, computed by the original (triangles) and modified (dots) algorithms. Blue markers correspond to АН samples, red markers — to АЛ, green — to ВН, and black — to ВЛ samples

пористости для 40 образцов и для обеих версий алгоритма (рис. 8) видна тенденция к снижению числа итераций для модифицированного алгоритма, причем эта тенденция является общей для всех образцов. Аналогично, на рис. 9 наблюдается увеличение времени на итерацию с ростом пористости для модифицированного алгоритма; здесь также отмечается общая тенденция для всех образцов. Возможно, это вызвано использованием операторов `if` в реализации алгоритма при вычислении по формуле (12). В таком случае дальнейшая оптимизация может повысить производительность. Для завершения сравнения двух версий алгоритма на рис. 10 приведено время решения уравнения Пуассона для всех 40 рассмотренных моделей с использованием каждой из версий. Видно, что при низкой пористости модифицированный алгоритм примерно на 20% быстрее исходного, а при увеличении пористости производительность модифицированного алгоритма возрастает примерно на 50–70% по сравнению с производительностью исходного алгоритма.

**6.4. Задачи большого размера.** Последняя серия экспериментов была проведена для иллюстрации применимости разработанного алгоритма к задачам крупномасштабного моделирования. Рассматривался цифровой образец С песчаника Бентхаймера, с помощью которого создавались кубические модели разных размеров. Поскольку размер исходного образца составлял  $400^3$  вокселей, то для получения образцов меньшего размера он разрезался на несколько частей. При этом образцы большего размера формировались путем тиражирования исходного образца с последующим вырезанием фрагментов требуемого размера. Таким образом, размеры образцов составляли  $(100n)^3$ , где  $n = 1, \dots, 10$ , т.е. наименьший образец составлял  $100^3$  вокселей, а наибольший —  $1000^3$  вокселей. Это был предельный размер для графическо-

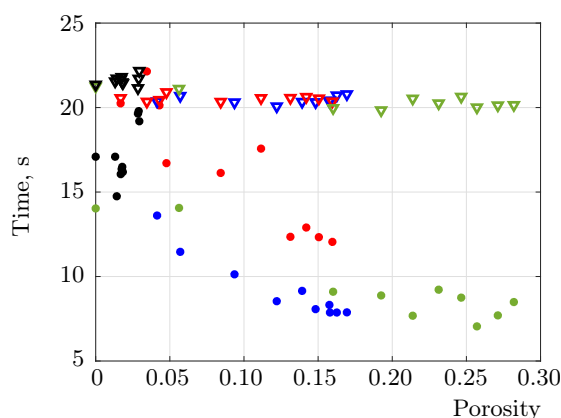


Рис. 10. Время решения уравнения Пуассона для 40 образцов карбонатов исходным (треугольники) и модифицированным (точки) алгоритмами. Синие маркеры соответствуют образцам АН, красные — АЛ, зеленые — ВН, черные — ВЛ

Fig. 10. Computational time for solving the Poisson equation for 40 samples of carbonates using the original (triangles) and the modified (dots) algorithms. Blue markers correspond to АН samples, red markers — to АЛ, green — to ВН, and black — to ВЛ samples



го процессора A100 с 80 ГБ памяти. При моделировании использовались следующие значения удельной электропроводности: 1 См/м для жидкости, заполняющей поры, и  $10^{-4}$  См/м для матрицы.

Уравнение Пуассона решалось с использованием двух версий алгоритма: исходной, где потенциал вычисляется во всех точках, и модифицированной, где решение строится только внутри порового пространства. Ниже приводятся графики числа итераций, времени работы алгоритма и времени на одну итерацию в зависимости от размера задачи, который варьировался от  $100^3 = 10^6$  до  $1000^3 = 10^9$  вокселей (рис. 11–13). Можно заметить, что в случае исходного алгоритма число итераций практически не зависит от размера задачи, за исключением наименьшей выборки. Напротив, увеличение сложности топологии порового пространства приводит к небольшому росту числа итераций для модифицированного алгоритма (рис. 11). Однако число итераций, необходимое для сходимости модифицированного алгоритма, меньше, чем у исходного. Время на одну итерацию практически одинаково для обоих алгоритмов и линейно растет с размером задачи (рис. 12). Наконец, время решения уравнения Пуассона с помощью модифицированного алгоритма составляет почти половину от времени, затрачиваемого исходным алгоритмом (рис. 13).

**7. Заключение.** В работе представлен численный алгоритм решения уравнения Пуассона применительно к задачам вычислительной физики горных пород и, в частности, к масштабированию удельной электропроводности (удельного электрического сопротивления) образцов горных пород, заполненных флюидом. Мы рассматриваем конечно-разностную аппроксимацию уравнения Пуассона и применяем метод сопряженных градиентов для решения полученной системы линейных алгебраических уравнений. Для построения предобуславливателя предлагается использовать обратный дискретизированный оператор Лапласа, соответствующий однородной среде. Действие предобуславливателя вычисляется с использованием

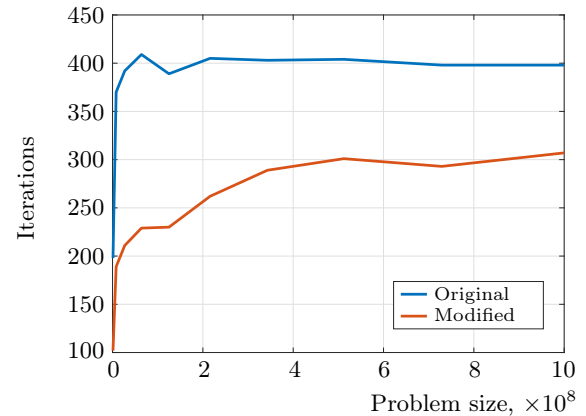


Рис. 11. Число итераций до достижения относительной невязки  $10^{-8}$  в зависимости от размера задачи. Синяя линия соответствует исходному алгоритму, красная — модифицированному

Fig. 11. Number of iterations required to achieve a relative residual of  $10^{-8}$  in dependence on the problem size. Blue line represents the original algorithm, red one corresponds to the modified algorithm

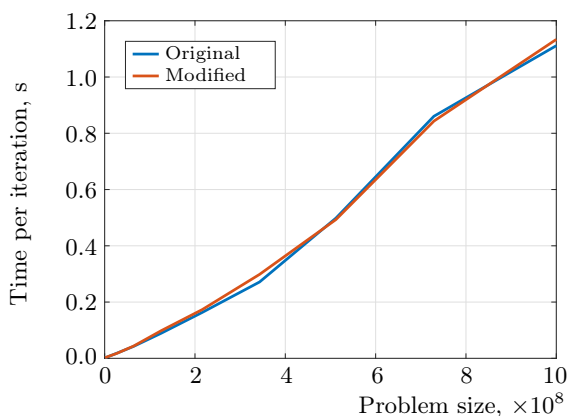


Рис. 12. Время одной итерации в зависимости от размера задачи. Синяя линия соответствует исходному алгоритму, красная — модифицированному

Fig. 12. Time per a single iteration in dependence on the problem size. Blue line represents the original algorithm, red one corresponds to the modified algorithm

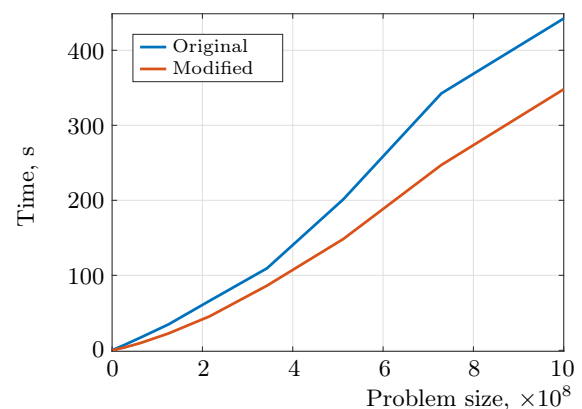


Рис. 13. Время решения уравнения Пуассона в зависимости от размера задачи. Синяя линия соответствует исходному алгоритму, красная — модифицированному

Fig. 13. Computational time for solving the Poisson equation in dependence on the problem size. Blue line represents the original algorithm, red one corresponds to the modified algorithm

ем одномерного спектрального матричного разложения (при помощи библиотек cuBLAS для вычисления матрично-матричных произведений) по двум направлениям и алгоритма Томаса для решения серии одномерных систем вдоль оставшегося направления.

Мы рассматриваем две версии алгоритма. В исходной версии уравнение Пуассона с высококонтрастными коэффициентами решается внутри всей области определения, т.е. в поровом пространстве и матрице. В модифицированной версии решение вычисляется только внутри порового пространства в предположении, что матрица является изолятором. В обоих случаях предложенный предобуславливатель работает эффективно. Тестирование алгоритма на реальных цифровых образцах горных пород размером до  $1000^3$  вокселей показало, что число итераций, необходимое для сходимости алгоритма, практически не зависит от размера задачи, но зависит от контраста проводимости жидкости в порах и матрицы породы. Более того, для модифицированного алгоритма сложность топологии порового пространства может влиять на скорость сходимости. Однако, тестирование показало, что для экстремальных контрастов проводимости (более  $10^4$ ) модифицированный алгоритм предпочтительнее, поскольку он почти вдвое быстрее исходного алгоритма.

Основным недостатком предлагаемого алгоритма является его ориентация на вычисления на одном графическом процессоре, что ограничивает его применимость задачами размером до  $1000^3$  вокселей на современных GPU. Поэтому основным направлением дальнейшего развития алгоритма является его адаптация к многопроцессорной архитектуре с использованием технологий MPI+CUDA или унифицированной памяти.

### Список литературы

1. Andrä H., Combaret N., Dvorkin J., et al. Digital Rock Physics Benchmarks — Part II: Computing Effective Properties // *Computers and Geosciences*. 2013. **50**. 33–43. doi [10.1016/j.cageo.2012.09.008](https://doi.org/10.1016/j.cageo.2012.09.008).
2. Bazaikin Y., Gurevich B., Iglaue S., Khachkova T., Kolyukhin D., Lebedev M., Lisitsa V., Reshetova G. Effect of CT image size and resolution on the accuracy of rock property estimates, *Journal of Geophysical Research // Solid Earth*. 2017. **122**, N. 5. 3635–3647. doi [10.1002/2016JB013575](https://doi.org/10.1002/2016JB013575)
3. Evstigneev N. M., Ryabkov O. I., Gerke K. M. Stationary stokes solver for single-phase flow in porous media: A blazingly fast solution based on algebraic multigrid method using GPU // *Advances in Water Resources*. 2023. **171**. 104340. doi [10.1016/j.advwatres.2022.104340](https://doi.org/10.1016/j.advwatres.2022.104340)
4. Khachkova T. S., Lisitsa V. V., Gondul E. A., Prokhorov D. I., Kostin V. I. Two-phase flow simulation algorithm for numerical estimation of relative phase permeability curves of porous materials // *Russian Journal of Numerical Analysis and Mathematical Modelling*. 2024. **39**, N. 4. 209–221. doi [10.1515/rnam-2024-0020](https://doi.org/10.1515/rnam-2024-0020).
5. Li M., Foroughi S., Zhao J., Bijeljic B., Blunt M. J. Image-based pore-scale modelling of the effect of wettability on breakthrough capillary pressure in gas diffusion layers // *Journal of Power Sources*. 2023. **584**. 233539. doi [10.1016/j.jpowsour.2023.233539](https://doi.org/10.1016/j.jpowsour.2023.233539).
6. Shulakova V., Pervukhina M., Müller T.M., et al. Computational Elastic Up-scaling of Sandstone on the Basis of X-Ray Micro-Tomographic Images // *Geophysical Prospecting*. 2012. **61**, N 2. 287–301. doi [10.1111/j.1365-2478.2012.01082.x](https://doi.org/10.1111/j.1365-2478.2012.01082.x).
7. 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](https://doi.org/10.26089/NumMet.v18r435).
8. Хачкова Т.С., Лисица В.В., Решетова Г.В., Чеверда В.А. Численная оценка удельного электрического сопротивления горных пород по их цифровым изображениям с использованием графических со-процессоров // *Вычислительные методы и программирование*. 2020. **21**, № 3. 306–318. doi [10.26089/NumMet.v21r326](https://doi.org/10.26089/NumMet.v21r326).
9. Zhan X., Schwartz L. M., Toksöz M. N., Smith W. C., Morgan F. D. Pore-scale modeling of electrical and fluid transport in berea sandstone // *Geophysics*. 2010. **75**, N. 5. F135–F142. doi [10.1190/1.3463704](https://doi.org/10.1190/1.3463704)
10. Dorn C., Schneider M. Lippmann-schwinger solvers for the explicit jump discretization for thermal computational homogenization problems // *International Journal for Numerical Methods in Engineering*. 2019. **118**, N 11. 631–653. doi [10.1002/nme.6030](https://doi.org/10.1002/nme.6030).
11. Molins S., Trebotich D., Steefel C. I., Shen C. An investigation of the effect of pore scale flow on average geochemical reaction rates using direct numerical simulation // *Water Resources Research*. 2012. **48**, N. 3. W03527. doi [10.1029/2011WR011404](https://doi.org/10.1029/2011WR011404)
12. Brown R. J. S. Connection between formation factor for electrical resistivity and fluid-solid coupling factor in biot's equations for acoustic waves in fluid-filled porous media // *Geophysics*. 1980. **45**, N. 8. 1269–1275. doi [10.1190/1.1441123](https://doi.org/10.1190/1.1441123).
13. Masson Y. J., Pride S. R., Nihei K. T. Finite difference modeling of Biot's poroelastic equations at seismic frequencies // *Journal of Geophysical Research: Solid Earth*. 2006. **111**, N. B10. 305. doi [10.1029/2006JB004366](https://doi.org/10.1029/2006JB004366)



14. Arns C. H., Knackstedt M. A., Mecke K. R. Characterisation of irregular spatial structures by parallel sets and integral geometric measures // Colloids and Surfaces A: Physicochemical and Engineering Aspects. 2004. **241**, N. 1-3. 351–372. doi [10.1016/j.colsurfa.2004.04.034](https://doi.org/10.1016/j.colsurfa.2004.04.034)
15. Хачкова Т. С., Базайкин Я. В., Лисица В. В. Применение методов вычислительной топологии для анализа изменения порового пространства породы в процессе химического растворения // Вычислительные методы и программирование. 2020. **21**, № 1. 41–55. doi [10.26089/NumMet.v21r104](https://doi.org/10.26089/NumMet.v21r104).
16. Saad, Y. Iterative Methods for Sparse Linear Systems. Philadelphia: Society for Industrial and Applied Mathematics, 2003. doi [10.1137/1.9780898718003](https://doi.org/10.1137/1.9780898718003).
17. Haber E., Ascher U.M., Aruliah D.A., Oldenburg D.W. Fast simulation of 3D electromagnetic problems using potentials // Journal of Computational Physics. 2000. **163**, N 1. 150–171. doi [10.1006/jcph.2000.6545](https://doi.org/10.1006/jcph.2000.6545).
18. Kostin V., Solov'yev S., Bakulin A., Dmitriev M. Direct frequency-domain 3D acoustic solver with intermediate data compression benchmarked against time-domain modeling for full-waveform inversion applications // Geophysics. 2019. **84**, N 4. 1–62. doi [10.1190/geo2018-0465.1](https://doi.org/10.1190/geo2018-0465.1).
19. Воронин К. В., Соловьев С. А. Решение уравнения Гельмгольца с использованием метода малоранговой аппроксимации в качестве предобуславливателя // Вычислительные методы и программирование. 2015. **16**, № 2. 268–280. doi [10.26089/NumMet.v16r226](https://doi.org/10.26089/NumMet.v16r226).
20. Johansen H., Colella P. A Cartesian grid embedded boundary method for Poisson's equation on irregular domains // Journal of Computational Physics. 1998. **147**, N 1. 60–85. doi [10.1006/jcph.1998.5965](https://doi.org/10.1006/jcph.1998.5965).
21. Stüben K. A review of algebraic multigrid // Journal of Computational and Applied Mathematics. 2001. **128**, N 1–2. 281–309. doi [10.1016/S0377-0427\(00\)00516-1](https://doi.org/10.1016/S0377-0427(00)00516-1).
22. Неклюдов Д. А., Сильвестров И. Ю., Чеверда В. А. Итерационный метод решения трехмерного уравнения Гельмгольца с “почти аналитическим” предобуславливателем для моделирования акустических волновых полей в задачах сейсморазведки // Вычислительные методы и программирование. 2014. **15**, № 3. 514–529. <https://num-meth.ru/index.php/journal/article/view/787/794>. (Дата обращения: 15 ноября 2025).
23. Belonosov M., Kostin V., Neklyudov D., Tcheverda V. 3D numerical simulation of elastic waves with a frequency-domain iterative solver // Geophysics. 2018. **83**, N 6. 333–344. doi [10.1190/geo2017-0710.1](https://doi.org/10.1190/geo2017-0710.1).
24. Khachkova T., Lisitsa V., Reshetova G., Tcheverda V. GPU-based algorithm for evaluating the electrical resistivity of digital rocks // Computers and Mathematics with Applications. 2021. **82**. 200–211. doi [10.1016/j.camwa.2020.11.005](https://doi.org/10.1016/j.camwa.2020.11.005).
25. Манаев А. А., Лисица В. В. Спектральный предобуславливатель для решения уравнения Пуассона // Вычислительные методы и программирование. 2025. **26**, N. 2. 111–128. doi [10.26089/NumMet.v26r208](https://doi.org/10.26089/NumMet.v26r208).
26. Hasbestan J. J., Xiao C.-N., Senocak I. Pittpack: An open-source Poisson's equation solver for extreme-scale computing with accelerators // Computer Physics Communications. 2020. **254**. 107272. doi [10.1016/j.cpc.2020.107272](https://doi.org/10.1016/j.cpc.2020.107272).
27. Самарский, А.А. Теория разностных схем. М.: Наука, 1983.
28. Vishnevsky D., Lisitsa V., Tcheverda V., Reshetova G. Numerical study of the interface errors of finite-difference simulations of seismic waves // Geophysics. 2014. **79**, N. 4. T219–T232. doi [10.1190/geo2013-0299.1](https://doi.org/10.1190/geo2013-0299.1)
29. Kim J. Phase-field models for multi-component fluid flows // Communications in Computational Physics. 2012. **12**, N. 3. 613–661. doi [10.4208/cicp.301110.040811a](https://doi.org/10.4208/cicp.301110.040811a)
30. 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 Resources Research. 2019. **55**, N. 4. 3171–3193. doi [10.1029/2018WR024137](https://doi.org/10.1029/2018WR024137)

Получена  
10 сентября 2025 г.

Принята  
16 октября 2025 г.

Опубликована  
19 ноября 2025 г.

### Информация об авторах

Алексей Андреевич Манаев — инженер; Институт математики имени С. Л. Соболева СО РАН, пр. Академика Коптюга, 4, 630090, Новосибирск, Российская Федерация.

Татьяна Станиславовна Хачкова — к.ф.-м.н., ст. научн. сотр.; Институт математики имени С. Л. Соболева СО РАН, пр. Академика Коптюга, 4, 630090, Новосибирск, Российская Федерация.

Вадим Викторович Лисица — д.ф.-м.н., заведующий лабораторией; Институт математики имени С. Л. Соболева СО РАН, пр. Академика Коптюга, 4, 630090, Новосибирск, Российская Федерация.

## References

1. H. Andrä, N. Combaret, J. Dvorkin, et al., “Digital Rock Physics Benchmarks — Part II: Computing Effective Properties,” *Computers and Geosciences*. **50**, 33–43 (2013). doi [10.1016/j.cageo.2012.09.008](https://doi.org/10.1016/j.cageo.2012.09.008).
2. Y. Bazaikin, B. Gurevich, S. Iglaue, T. Khachkova, D. Kolyukhin, M. Lebedev, V. Lisitsa, G. Reshetova, “Effect of CT image size and resolution on the accuracy of rock property estimates,” *Journal of Geophysical Research. Solid Earth*. **122**, (5), 3635–3647 (2017). doi [10.1002/2016JB013575](https://doi.org/10.1002/2016JB013575)
3. N. M. Evstigneev, O. I. Ryabkov, K. M. Gerke, “Stationary stokes solver for single-phase flow in porous media: A blasting fast solution based on algebraic multigrid method using GPU,” *Advances in Water Resources*. **171**, 104340 (2023). doi [10.1016/j.advwatres.2022.104340](https://doi.org/10.1016/j.advwatres.2022.104340)
4. T. S. Khachkova, V. V. Lisitsa, E. A. Gondul, D. I. Prokhorov, V. I. Kostin, “Two-phase flow simulation algorithm for numerical estimation of relative phase permeability curves of porous materials,” *Russian Journal of Numerical Analysis and Mathematical Modelling* **39**, (4), 209–221 (2024). doi [10.1515/rnam-2024-0020](https://doi.org/10.1515/rnam-2024-0020).
5. M. Li, S. Foroughi, J. Zhao, B. Bijeljic, M. J. Blunt, “Image-based pore-scale modelling of the effect of wettability on breakthrough capillary pressure in gas diffusion layers,” *Journal of Power Sources*, **584**, 233539 (2023). doi [10.1016/j.jpowsour.2023.233539](https://doi.org/10.1016/j.jpowsour.2023.233539).
6. V. Shulakova, M. Pervukhina, T. M. Müller, et al., “Computational Elastic Up-scaling of Sandstone on the Basis of X-Ray Micro-Tomographic Images,” *Geophysical Prospecting*. **61** (2), 287–301 (2012). doi [10.1111/j.1365-2478.2012.01082.x](https://doi.org/10.1111/j.1365-2478.2012.01082.x).
7. G. V. Reshetova and T. 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).
8. T. S. Khachkova, V. V. Lisitsa, G. V. Reshetova, and V. A. Tcheverda, “Numerical estimation of electrical resistivity in digital rocks using GPUs,” *Numerical Methods and Programming*. **21** (3), 306–318 (2020). doi [10.26089/NumMet.v21r326](https://doi.org/10.26089/NumMet.v21r326).
9. X. Zhan, L. M. Schwartz, M. N. Toksöz, W. C. Smith, F. D. Morgan, “Pore-scale modeling of electrical and fluid transport in berea sandstone,” *Geophysics*, **75**, (5), F135–F142 (2010). doi [10.1190/1.3463704](https://doi.org/10.1190/1.3463704)
10. C. Dorn and M. Schneider, “Lippmann-schwinger solvers for the explicit jump discretization for thermal computational homogenization problems,” *International Journal for Numerical Methods in Engineering*. **118** (11), 631–653 (2019). doi [10.1002/nme.6030](https://doi.org/10.1002/nme.6030).
11. S. Molins, D. Trebotich, C. I. Steefel, C. Shen, “An investigation of the effect of pore scale flow on average geochemical reaction rates using direct numerical simulation,” *Water Resources Research*, **48**, (3), W03527 (2012). doi [10.1029/2011WR011404](https://doi.org/10.1029/2011WR011404)
12. R. J. S. Brown, “Connection between formation factor for electrical resistivity and fluid-solid coupling factor in Biot’s equations for acoustic waves in fluid-filled porous media,” *Geophysics*, **45**, (8), 1269–1275 (1980). doi [10.1190/1.1441123](https://doi.org/10.1190/1.1441123).
13. Y. J. Masson, S. R. Pride, K. T. Nihei, “Finite difference modeling of Biot’s poroelastic equations at seismic frequencies,” *Journal of Geophysical Research: Solid Earth*, **111**, (B10), 305 (2006). doi [10.1029/2006JB004366](https://doi.org/10.1029/2006JB004366)
14. C. H. Arns, M. A. Knackstedt, K. R. Mecke, “Characterisation of irregular spatial structures by parallel sets and integral geometric measures,” *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, **241**, (1-3), 351–372 (2004). doi [10.1016/j.colsurfa.2004.04.034](https://doi.org/10.1016/j.colsurfa.2004.04.034)
15. T. S. Khachkova, Y. V. Bazaikin, and V. V. Lisitsa, “Use of the computational topology to analyze the pore space changes during chemical dissolution,” *Numerical Methods and Programming*. **21** (1), 41–55 (2020). doi [10.26089/NumMet.v21r104](https://doi.org/10.26089/NumMet.v21r104).
16. Y. Saad, *Iterative Methods for Sparse Linear Systems* (Society for Industrial and Applied Mathematics, Philadelphia, 2003) [in Russian]. doi [10.1137/1.9780898718003](https://doi.org/10.1137/1.9780898718003).
17. E. Haber, U. M. Ascher, D. A. Aruliah, and D. W. Oldenburg, “Fast simulation of 3D electromagnetic problems using potentials,” *Journal of Computational Physics*. **163** (1), 150–171 (2000). doi [10.1006/jcph.2000.6545](https://doi.org/10.1006/jcph.2000.6545).
18. V. Kostin, S. Solovyev, A. Bakulin, and M. Dmitriev, “Direct frequency-domain 3D acoustic solver with intermediate data compression benchmarked against time-domain modeling for full-waveform inversion applications,” *Geophysics*. **84** (4), 1–62 (2019). doi [10.1190/geo2018-0465.1](https://doi.org/10.1190/geo2018-0465.1).
19. K. V. Voronin and S. A. Solovyev, “Solution of the Helmholtz problem using the preconditioned low-rank approximation technique,” *Numerical Methods and Programming*. **16** (2), 268–280 (2015). doi [10.26089/NumMet.v16r226](https://doi.org/10.26089/NumMet.v16r226).



20. H. Johansen and P. Colella, “A Cartesian grid embedded boundary method for Poisson’s equation on irregular domains,” *Journal of Computational Physics*. **147** (1), 60–85 (1998). doi [10.1006/jcph.1998.5965](https://doi.org/10.1006/jcph.1998.5965).
21. K. Stüben, “A review of algebraic multigrid,” *Journal of Computational and Applied Mathematics*. **128** (1–2), 281–309 (2001). doi [10.1016/S0377-0427\(00\)00516-1](https://doi.org/10.1016/S0377-0427(00)00516-1).
22. D. A. Neklyudov, I. Yu. Silvestrov, and V. A. Tcheverda, “A 3D Helmholtz iterative solver with a semi-analytical preconditioner for acoustic wavefield modeling in seismic exploration problems,” *Numerical Methods and Programming*. **15** (3), 514–529 (2014). <https://num-meth.ru/index.php/journal/article/view/787/794> [in Russian]. Cited November 15, 2025.
23. M. Belonosov, V. Kostin, D. Neklyudov, and V. Tcheverda, “3D numerical simulation of elastic waves with a frequency-domain iterative solver,” *Geophysics*. **83** (6), 333–344 (2018). doi [10.1190/geo2017-0710.1](https://doi.org/10.1190/geo2017-0710.1).
24. T. Khachkova, V. Lisitsa, G. Reshetova, and V. Tcheverda, “GPU-based algorithm for evaluating the electrical resistivity of digital rocks,” *Computers and Mathematics with Applications*. **82**, 200–211 (2021). doi [10.1016/j.camwa.2020.11.005](https://doi.org/10.1016/j.camwa.2020.11.005).
25. A. A. Manaev and V. V. Lisitsa, “Spectral preconditioner for solving the Poisson equation,” *Numerical Methods and Programming*, **26**, 2, 111–128 (2025). doi [10.26089/NumMet.v26r208](https://doi.org/10.26089/NumMet.v26r208).
26. J. J. Hasbestan, C.-N. Xiao, I. Senocak, “Pittpack: An open-source Poisson’s equation solver for extreme-scale computing with accelerators,” *Computer Physics Communications*, **254**, 107272 (2020). doi [10.1016/j.cpc.2020.107272](https://doi.org/10.1016/j.cpc.2020.107272).
27. A. A. Samarskii, *The theory of difference schemes* (Nauka, Moscow, 1983) [in Russian].
28. D. Vishnevsky, V. Lisitsa, V. Tcheverda, G. Reshetova, “Numerical study of the interface errors of finite-difference simulations of seismic waves,” *Geophysics*, **79**, (4), T219–T232 (2014). doi [10.1190/geo2013-0299.1](https://doi.org/10.1190/geo2013-0299.1)
29. J. Kim, “Phase-field models for multi-component fluid flows,” *Communications in Computational Physics* **12**, (3), 613–661 (2012). doi [10.4208/cicp.301110.040811a](https://doi.org/10.4208/cicp.301110.040811a)
30. Y. Al-Khulaifi, Q. Lin, M. J. Blunt, B. Bijeljic, “Pore-scale dissolution by CO<sub>2</sub> saturated brine in a multimineral carbonate at reservoir conditions: Impact of physical and chemical heterogeneity,” *Water Resources Research*, **55**, (4), 3171–3193 (2019). doi [10.1029/2018WR024137](https://doi.org/10.1029/2018WR024137)

Received  
September 10, 2025

Accepted  
October 16, 2025

Published  
November 19, 2025

### Information about the authors

*Aleksei A. Manaev* — Engineer; Sobolev Institute of Mathematics SB RAS, Koptyug prospekt, 4, 630090, Novosibirsk, Russia.

*Tatyana S. Khachkova* — Ph. D., Senior Researcher; Sobolev Institute of Mathematics SB RAS, Koptyug prospekt, 4, 630090, Novosibirsk, Russia.

*Vadim V. Lisitsa* — Dr. Sci., Head of Laboratory; Sobolev Institute of Mathematics SB RAS, Koptyug prospekt, 4, 630090, Novosibirsk, Russia.