

ДИНАМИЧЕСКИЕ СИСТЕМЫ И ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

DYNAMIC SYSTEMS AND OPTIMAL CONTROL



Серия «Математика»
2026. Т. 57. С. 3–18

Онлайн-доступ к журналу:
<http://mathizv.isu.ru>

ИЗВЕСТИЯ

Иркутского
государственного
университета

Научная статья

УДК 517.97

MSC 49J35, 65K10

DOI <https://doi.org/10.26516/1997-7670.2026.57.3>

Тензорное представление численного метода решения задачи Плато

В. А. Клячин^{1,2✉}

¹ Волгоградский государственный университет, Волгоград, Российская Федерация

² Новосибирский государственный университет, Новосибирск, Российская
Федерация

✉ klchnv@mail.ru

Аннотация: Предлагается операторный подход для решения вычислительных задач на нерегулярных сетках в виде триангуляций. В основе предлагаемого метода лежит организация вычислительного процесса как последовательности операций над многомерными тензорами, которые представляют все обрабатываемые данные, включая описание структуры триангуляции, вершины триангуляции, стороны треугольников и нормали к ним, а также значения сеточных функций в вершинах триангуляции. Для демонстрации эффективности предлагаемого метода проведен ряд численных экспериментов по вычислению градиента кусочно-линейной функции на триангуляциях. Сравнение времени приводилось с прямым методом вычисления, который подразумевает выполнение вычислений для каждого отдельного треугольника. В качестве нетривиального примера применения указанного подхода рассматривается численное решение задачи Плато на поиск полигональной поверхности минимальной площади в $\mathbb{R}^3$ с фиксированным граничным контуром в виде пространственной ломаной. Решение этой задачи ищется методом градиентного спуска. Расчетная схема представлена в виде тензорных операций, список которых приводится. Программная реализация выполнена с использованием языка программирования Python и его библиотеки NumPy, предназначенной для работы с многомерными массивами. Приведены примеры расчета приближенных решений задачи на минимум площади, моделирующие известные минимальные поверхности: геликоид, катеноид и поверхность Шерка.

Ключевые слова: триангуляция, полигональная поверхность, задача на минимум площади, численные методы, тензорные операции

Благодарности: Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования Российской Федерации № 075-15-2025-349 от 29.04.2025

Ссылка для цитирования: Клячин В. А. Тензорное представление численного метода решения задачи Плато // Известия Иркутского государственного университета. Серия Математика. 2026. Т. 57. С. 3–18.
<https://doi.org/10.26516/1997-7670.2026.57.3>

Research article

Tensor Representation of a Numerical Method for Solving the Plateau Problem

Vladimir A. Klyachin^{1,2✉}

¹ Volgograd State University, Volgograd, Russian Federation

² Novosibirsk State University, Novosibirsk, Russian Federation

✉ klchnv@mail.ru

Abstract: This paper proposes an operator approach for solving computational problems on irregular meshes in the form of triangulations. The proposed method is based on organizing the computational process as a sequence of operations on multidimensional tensors that represent all the processed data, including a description of the triangulation structure, triangulation vertices, triangle sides and normals to them, and the values of mesh functions at the triangulation vertices. To demonstrate the effectiveness of the proposed method, the author conducted a series of numerical experiments on calculating the gradient of a piecewise linear function on triangulations. A comparison of the time was given with the direct calculation method, which involves performing calculations for each individual triangle. As a nontrivial example of the application of this approach, this paper considers the numerical solution of the Plateau problem of finding a polygonal surface of minimum area in $\mathbb{R}^3$ with a fixed boundary contour in the form of a spatial polygonal line. The solution to this problem is sought using the gradient descent method. The computational scheme is presented in the form of tensor operations, a list of which is given in the paper. The software implementation is implemented using the Python programming language and its NumPy library, designed for working with multidimensional arrays. Examples of calculating approximate solutions to the area minimum problem are provided, modeling well-known minimal surfaces: the helicoid, the catenoid, and the Scherk surface.

Keywords: triangulation, polygonal surface, area minimum problem, numerical methods, tensor operations

Acknowledgements: The work is supported by the Mathematical Center in Akademgorodok, the agreement with the Ministry of Science and Higher Education of the Russian Federation No. 075-15-2025-349 from 29.04.2025.

For citation: Klyachin V. A. Tensor Representation of a Numerical Method for Solving the Plateau Problem. *The Bulletin of Irkutsk State University. Series Mathematics*, 2026, vol. 57, pp. 3–18. (in Russian)
<https://doi.org/10.26516/1997-7670.2026.57.3>

1. Введение

В работах [7; 8] предложена операторная форма представления разностных схем для решения задач математической физики. Такое представление дано в терминах разностных (дискретных) аналогов дифференциальных операторов div , grad , rot . При этом для аппроксимаций этих операторов требовалось условие выполнения соответствующих аналогов физических законов сохранения, которые в непрерывном случае записываются в интегральной форме. В указанных статьях решение поставленной задачи дано на регулярных косоугольных пространственных сетках, узлы которых индексируются тремя натуральными числами. В работе [5] также решалась задача об операторном представлении расчетных схем на основе некоторого набора сеточных операторов. Там же была отмечена важность такого рода представлений для упрощения записи разностных схем средствами языков программирования, ускорения вычислений, а также в связи с развитием параллельных вычислений на базе технологии CUDA для графических процессоров NVIDIA. Отметим, что в этой статье, так же как и в вышеупомянутых, рассматриваются сеточные функции на регулярных пространственных сетках.

Основываясь на идеях указанных выше работ, мы в настоящей статье описываем аналогичный подход для организации вычислений на триангуляциях как примере нерегулярных сеток. При этом структура триангуляции, ее вершины, ребра и треугольные грани, а также значения сеточных функций представлены в виде многомерных тензоров, а вычислительный процесс может быть записан в виде последовательности операций над этими тензорами. В рамках данной вычислительной модели мы рассматриваем численное решение задачи Плато [3] в классе полигональных поверхностей. Задача заключается в поиске поверхности минимальной площади с заданным граничным контуром в виде пространственной ломаной [12]. Ускорение вычислений достигается за счет того, что приводимые ниже в статье тензорные операции эффективно реализованы, например, в библиотеке NumPy. Автором проведен ряд численных экспериментов по вычислению градиента кусочно-линейной функции на триангуляциях для сравнения прямого метода вычисления с методом на основе тензорных операций. Результаты замеров времени показывают, что в среднем вычисления градиента методом тензорных операций быстрее прямого метода в 55 раз.

2. Тензорные операции

Здесь мы описываем набор тензорных операций, в терминах которых планируется выражать численные расчеты вычислительной схемы. Рассмотрим кортеж из n натуральных чисел $k_1, k_2, \dots, k_n$, $n > 0$ и определим векторное пространство тензоров $V(k_1, k_2, \dots, k_n)$ как элементов $T = \{T_{i_1 i_2 \dots i_n}\}$, содержащих наборы чисел $T_{i_1 i_2 \dots i_n}$ $1 \leq i_j \leq k_j$, $j = 1, 2, \dots, n$. В этом пространстве естественным образом определены операции покомпонентного сложения и умножения на число.

Определим следующие операции над тензорами:

- Линейные операции — покомпонентные сложение тензоров и умножение их на число (в том числе комплексное).
- Операция свертки $*$, включающая скалярное произведение векторов и умножение матриц. Рассмотрим пример. Пусть $A \in V(k_1, k_2, \dots, k_n)$ и $B \in V(l_1, l_2, \dots, l_m)$, $k_n = l_{m-1}$ — два тензора (две многомерные матрицы), тогда

$$(A * B)_{i_1 i_2 \dots i_{n-1} j_1 j_2 \dots j_{m-2} j_m} = \sum_k A_{i_1 i_2 \dots i_{n-1} k} B_{j_1 j_2 \dots j_{m-2} k j_m}, \quad (2.1)$$

где индекс k у матрицы берется по последней размерности, а у B — по предпоследней. В результате получаем тензор $A * B \in V(k_1, \dots, k_{n-1}, l_1, \dots, l_{m-2}, l_m)$. Данная операция часто используется в алгоритмах обучения нейронных сетей, алгоритмах компьютерного зрения и др. При этом индекс k у обоих тензоров должен изменяться в одном диапазоне.

- Суммирование элементов вдоль заданной размерности

$$\text{sum}_j(A) = \sum_{i=1}^{k_j} A_{i_1 i_2 \dots i_{j-1} i_{j+1} \dots i_n}. \quad (2.2)$$

- Поэлементные арифметические операции: $+$, $\cdot$, $/$, $-$ тензоров одинаковой размерности.
- Пусть $A \in V_{k_1, k_2, \dots, k_n}$, $j \in \{1, 2, \dots, n\}$ и $\sigma_j : \{1, 2, \dots, k_j\} \rightarrow \{1, 2, \dots, k_j\}$ — некоторая перестановка. Определим операцию перестановки тензора

$$\sigma_j(A)_{i_1 \dots i_j \dots i_n} = A_{i_1 \dots \sigma_j(i_j) \dots i_n}. \quad (2.3)$$

Отдельно рассмотрим операции индексирования. Пусть даны тензор $A \in V(k_1, k_2, \dots, k_n)$ и $I = (i_1, i_2, \dots, i_N)$ — произвольный набор номеров $1 \leq i_j \leq k_1$. Тогда сформируем новый тензор

$$B = A[I], \quad B \in V(N, k_2, \dots, k_n),$$

полагая

$$B_{j_1 j_2 \dots j_N} = A_{I[j_1] j_2 \dots j_N}, \quad j_1 = i_1, i_2, \dots, i_N. \quad (2.4)$$

Другими словами, выбираются компоненты тензора A , первый индекс у которых берется равным значению из вектора I . Обобщенным вариантом данной операции индексации является следующая. Рассмотрим произвольный целочисленный тензор $I \in V(l_1, \dots, l_m)$, причем $1 \leq I_{j_1 j_2 \dots j_m} \leq k_1$. Определим тензор $B = A[I]$, $B \in V(l_1, \dots, l_m, k_2, \dots, k_n)$, компоненты которого вычисляются по формуле

$$B_{j_1 j_2 \dots j_m, i_2, \dots, i_n} = A_{I_{j_1 j_2 \dots j_m} i_2 \dots i_n}. \quad (2.5)$$

Пусть тензор $A \in V(k_1, k_2, \dots, k_n)$. Предположим, что некоторый номер $k_j = 1$. Тогда тензор A включается в пространства

$$V(k_1, \dots, k_{j-1}, k, k_{j+1}, \dots, k_n)$$

для всякого натурального $k \geq 1$ с помощью операции *наращивания*, в результате которой получается тензор $A' \in V(k_1, \dots, k_{j-1}, k, k_{j+1}, \dots, k_n)$, компоненты которого вычисляются по формуле

$$A'_{i_1 i_2 \dots i_{j-1} i_j i_{j+1} \dots i_n} = A_{i_1 i_2 \dots i_{j-1} 1 i_{j+1} \dots i_n}.$$

Такую операцию наращивания можно применить одновременно к нескольким индексам, у которых диапазон ограничен одним значением 1. Далее, если $A \in V(k_1, k_2, \dots, k_n)$, то этот тензор включается в пространство $V(k_1, \dots, k_n, 1)$ с помощью аналогичной операции наращивания

$$A'_{i_1 i_2 \dots i_n 1} = A_{i_1 i_2 \dots i_n}.$$

Тензорную бинарную операцию для двух тензоров $A \in V_{k_1, k_2, \dots, k_n}$ и $B \in V(l_1, l_2, \dots, l_m)$, $m \leq n$ назовем допустимой, если для всякого $j = 1, \dots, m$ выполнено одно из трех равенств

$$l_j = k_j, \quad k_j = 1, \quad l_j = 1,$$

а для $j = m+1, \dots, n$ выполнено $k_j = 1$. Обозначим произвольно выбранную бинарную тензорную операцию через $\odot$. При выполнении условия допустимости этой операции полагаем

$$(A \odot B)_{s_1 s_2 \dots s_n} = A'_{s_1 s_2 \dots s_n} \odot B'_{s_1 s_2 \dots s_n}.$$

Рассмотрим пример. Пусть матрица P в каждой строке содержит координаты точек в пространстве $\mathbb{R}^3$ и пусть имеется некоторый вектор $q \in \mathbb{R}^3$. Тогда операция $P+q$ допустима и, согласно вышеприведенным формулам, вычисляет матрицу, строки которой являются суммами строк матрицы P и вектора q . Другими словами, эта операция производит построчное суммирование матрицы P с вектором q .

Наконец, рассмотрим следующие операторы проекции. Выберем некоторый номер $1 \leq j \leq n$ и два натуральных числа $1 \leq k \leq l \leq k_j$. Рассмотрим оператор

$$S_{k,l}^j : V(k_1, \dots, k_j, \dots, k_n) \rightarrow V(k_1, \dots, l-k+1, \dots, k_n)$$

Такой, что

$$(S_{k,l}^j(T))_{i_1 \dots i_j \dots i_n} = T_{i_1 \dots k+i_j-1 \dots i_n}. \quad (2.6)$$

Так что S_{1,k_j}^j — тождественный оператор $S_{1,k_j}^j(T) = T$. Заметим, что вышеприведенные операции реализованы в пакете NumPy языка программирования Python. Мы используем этот пакет для демонстрации результатов расчетов на основе тензорных операций.

3. Формулировка задачи на минимум функционала типа площади

Равновесные поверхности как поверхности раздела двух сред, находящихся в механическом равновесии, играют важную роль в науке и технике [9]. Одним из классов такого рода поверхностей является класс капиллярных поверхностей. Математическая теория капиллярных поверхностей подробно рассмотрена в [10]. Частным случаем капиллярных поверхностей является класс поверхностей, минимизирующих функционал площади, — класс минимальных поверхностей. Для моделирования таких поверхностей произвольной топологии мы обратимся к общему случаю функционалов типа площади. Пусть $\Phi : \mathbb{R}^3 \rightarrow \mathbb{R}$ — положительная однородная степени 1, выпуклая функция такая, что $\Phi(\xi) = \Phi(-\xi)$. Для гладкой поверхности $\mathcal{M} \subset \mathbb{R}^3$ рассмотрим величину

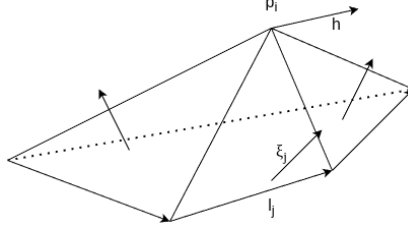
$$F(\mathcal{M}) = \int_{\mathcal{M}} \Phi(\xi) dS,$$

где ξ — вектор единичной нормали к поверхности $\mathcal{M}$.

Будем говорить, что поверхность $\mathcal{M}'$ является кусочно-линейной аппроксимацией поверхности $\mathcal{M}$, если $\mathcal{M}'$ является полигональной поверхностью с треугольными гранями и ее вершины лежат на поверхности $\mathcal{M}$. Если $\mathcal{M}'$ — кусочно-линейная аппроксимация поверхности $\mathcal{M}$, то приближенное значение введенного функционала будет вычисляться согласно формуле

$$F(\mathcal{M}') = \sum_{i=1}^M \Phi(\xi_i) |\Delta_i|,$$

где $\mathcal{M}'$ представляет собой объединение треугольников Δ_i , ξ_i — нормаль к плоскости этих треугольников, а $|\Delta_i|$ — площадь треугольников. Поверхность $\mathcal{M}'$ можно рассматривать как точку в пространстве $\mathbb{R}^{3N}$ размерности $3N$, где N — число вершин на $\mathcal{M}'$. Если задать в каждой точке-вершине поверхности вектор h , то этим векторам будет соответствовать вектор в $\mathbb{R}^{3N}$. Тогда оказывается, если функционал F


 Рис. 1. Вычисление $\frac{\partial F}{\partial h}(p_i)$

рассмотреть как числовую функцию $F : \mathbb{R}^{3N} \rightarrow \mathbb{R}$, что имеет место формула

$$\frac{\partial F}{\partial h}(p_i) = \frac{1}{2} \left\langle \sum_{j=1}^k \left(\nabla \Phi(\xi_j) |\xi_j| + \frac{\xi_j}{|\xi_j|} \Phi(\xi_j) \right) \times l_j, h \right\rangle, \quad (3.1)$$

где сумма идет по всем треугольникам, имеющим вершиной точку $p_i \in \mathcal{M}'$, ξ_j обозначает нормаль к соответствующему треугольнику и l_j — вектор, соединяющий вершины треугольника, не совпадающие с p_i таким образом, что эта вершина остается слева (рис. 1). Рассмотрим случай $\Phi(\xi) = |\xi|$. Тогда, как нетрудно посчитать, получим

$$\frac{\partial F}{\partial h}(p_i) = \left\langle \sum_{j=1}^k \xi_j \times l_j, h \right\rangle. \quad (3.2)$$

Если предположить, что поверхность $\mathcal{M}'$ минимизирует функционал $F(\mathcal{M}')$, то

$$\frac{\partial F}{\partial h}(p_i) = \left\langle \sum_{j=1}^k \xi_j \times l_j, h \right\rangle = 0$$

для любой вершины $p_i \in \mathcal{M}'$.

Для минимизации функционалов рассматриваемого вида применим метод градиентного спуска. Рассмотрим некоторую кусочно-линейную поверхность $\mathcal{M}_0$. Метод градиентного спуска предполагает на каждом шаге итерации получения очередной поверхности $\mathcal{M}_k$ согласно формуле

$$\mathcal{M}_k = \mathcal{M}_{k-1} - \varepsilon_k \nabla F,$$

где равенство рассматривается как векторное соотношение в пространстве $\mathbb{R}^{3N}$, а числа ε_k будем определять экспериментально. При этом граничные точки остаются неподвижными. Это можно обеспечить, если мы положим $\nabla F(q) = 0$ в каждой вершине полигональной поверхности

$\mathcal{M}'$, являющейся граничной. Для вычисления градиента ∇F воспользуемся матричными операциями, приведенными в предыдущем разделе статьи.

4. Вычисление нормалей полигональной поверхности

Для организации вычисления градиента функционала площади в соответствии с формулой (3.2) обратимся к задаче вычисления нормалей для полигональной поверхности. Пусть поверхность $\mathcal{M}$ задана треугольной сеткой, т. е. она представляет собой пространственную многогранную поверхность с треугольными гранями. Пусть N — число ее вершин, а M — число треугольников. Тогда $\mathcal{M}$ задается парой матриц P и T . В строках первой матрицы размещаются координаты вершин так, что $P \in V(N, 3)$, а в строках матрицы T расположены индексы вершин соответствующего треугольника. Поэтому $T \in V(M, 3)$. Как было показано выше, вычисление градиента предполагает вычисление вектора нормали к каждой грани поверхности. Представим это вычисление в матричном виде. Именно, рассмотрим 3-мерный тензор J_{ijk} такой, что

$$J_{1jk} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad J_{2jk} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}, \quad J_{3jk} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Несложно убедиться, что для всякой пары векторов $a, b \in \mathbb{R}^3$, согласно (2.1), выполнено $a \times b = (J * a) \cdot b$. Если поверхность задана парой матриц $\mathcal{M} = (P, T)$, как описано выше, то матрица V с координатами нормалей граней вычисляется следующим образом:

Шаг 1. Первое действие состоит в том, чтобы получить тензор Q , содержащий для каждой грани поверхности координаты всех ее трех вершин. Согласно правилам индексации (2.5), имеем $Q = P[T]$. Применяя операторы проекции (2.6), вычислим тензор, содержащий пару векторов каждой грани $R = S_{2,3}^2(Q) - S_{1,1}^2(Q)$. Еще раз применяя операторы проекции, выделим эти векторы в отдельные тензоры $A = S_{2,2}^2(R), B = S_{1,1}^2(R)$.

Шаг 2. Далее, в соответствии с формулой (2.2), вычислим для каждой грани векторное произведение этих векторов в два действия: $U = \text{sum}_3(A \cdot J), \quad V = \text{sum}_2(B \cdot U)$. Для последующих вычислений введем операцию

$$\text{cross}(A, B) = \text{sum}_2(B \cdot \text{sum}_3(A \cdot J))$$

для двух матриц $A, B \in V(N, 3)$. Можно проверить выполнение равенства $\text{sum}_3(V \cdot R) = 0$. Здесь справа стоит 0 как тензор пространства $V(M, 2)$. Это равенство означает ортогональность векторов из V векторам из R .

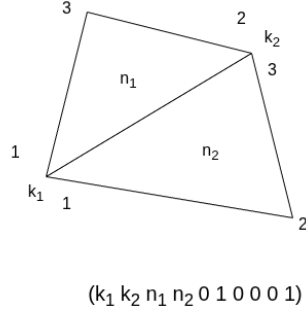


Рис. 2. Иллюстрация к описанию строки матрицы H

Шаг 3. Теперь нормируем векторы из тензора V , применяя (2.2):

$$L = \sqrt{\text{sum}_2(V \cdot V)}, \quad V = V/L.$$

5. Матрицы, описывающие структуру триангуляции

Для векторного представления схемы вычисления градиента функционала площади нам понадобятся вспомогательные матрицы. Пронумеруем все стороны всех треугольников граней поверхности. Первая матрица — матрица инцидентности реберного графа несущей сетки триангуляции поверхности

$$I_{ij} = \begin{cases} 1, & \text{если вершина с номером } i \text{ и } j\text{-я сторона} \\ & \text{инцидентны, т. е. } i\text{-я вершина} \\ & \text{является концом отрезка } j\text{-й стороны} \\ 0 & \text{в противном случае.} \end{cases}$$

Следующая матрица будет содержать описание соседних связей треугольников и ребер. Обозначим эту матрицу через H . Каждая ее строка содержит 10 неотрицательных целых чисел $k_1, k_2, n_1, n_2, b_1, b_2, b_3, c_1, c_2, c_3$ и соответствует одному ребру какой-либо треугольной грани (рис. 2). Первые два ее элемента k_1, k_2 равны номерам вершин, являющихся концами соответствующего ребра. Следующие два числа n_1, n_2 равны номерам треугольников, лежащих по разные стороны от прямой отрезка этого ребра.

Следующие две группы троек из нулей и единиц $b_i, c_i \in \{0, 1\}$ указывают, напротив какой вершины лежит общая сторона этих двух треугольников. Например, если $(b_1, b_2, b_3) = (0, 1, 0)$, то это значит, что ребро k_1, k_2 в треугольнике с номером n_1 лежит напротив вершины с номером 2 этого треугольника, а если $(c_1, c_2, c_3) = (0, 0, 1)$, то это значит,

что ребро k_1, k_2 в треугольнике с номером n_2 лежит напротив вершины с номером 3 этого треугольника. Для граничных ребер все числа равны 0 — для отсутствующего треугольника. Указанные две матрицы вычисляются один раз и не требуют обновлений на каждой итерации градиентного спуска для минимизации функционала площади.

6. Вычисление градиента функционала площади

Мы опишем схему вычисления градиента по формуле (3.2). Общий случай (3.1) немногим отличается от рассматриваемого. Итак, предположим, что вычислен тензор U_{jsk} , элементы которого равны k -й координате вектора, лежащего в j -й грани и ортогональной стороне, лежащей напротив s -й вершины грани (рис. 3). Тогда можно вычислить матрицу $W_{ik} \in V(K, 3)$, где K — число ребер полигональной поверхности. Значение W_{ik} равно k -й координате суммы двух векторов из U , ортогональных общей стороне. Тогда матрица G градиента площади легко вычисляется по формуле

$$G = I \cdot U. \quad (6.1)$$

Теперь представим весь процесс вычисления по шагам.

Шаг 1. Рассмотрим перестановку $\sigma_2 : \{1, 2, 3\} \mapsto \{2, 3, 1\}$. Пусть $T^1 = \sigma_2(T)$, $T^2 = \sigma_2(T^1)$. Тогда тензор $R = P[T^2] - P[T^1]$ принадлежит $V(M, 3, 3)$ и содержит координаты векторов, направленных вдоль сторон каждой грани так, что R_{jsk} есть k -я координата вектора, соединяющего вершины j -й грани, номера которых не равны номеру s (рис. 3). Теперь тензор $U \in V(M, 3, 3)$ может быть вычислен по формуле $U = \text{cross}(R, V)$, где тензор V содержит нормали к каждой грани полигональной поверхности.

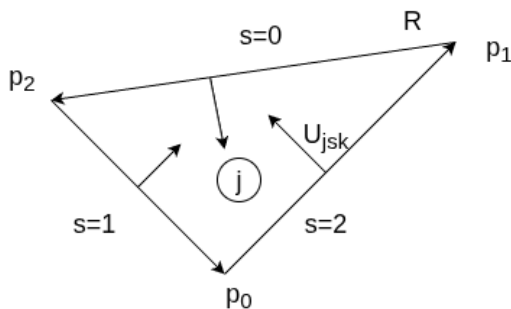
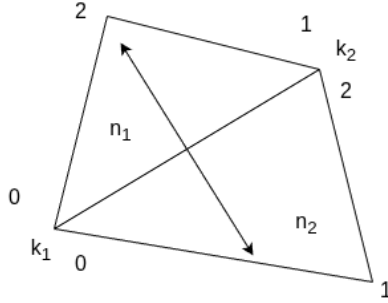


Рис. 3. Иллюстрация к вычислению тензора U

Таким образом мы вычисляем матрицу, состоящую из векторов, ортогональных сторонам треугольных граней.


 Рис. 4. Иллюстрация к вычислению тензора X

Шаг 2. После этого, используя матрицы H, I , описывающие структуру триангуляции поверхности, вычисляем векторы, ортогональные сторонам смежных граней, т. е. граней, имеющих общее ребро. Для этого используем оператор проекции (2.6) и правило индексирования (2.5):

$$Z = S_{3,4}^2(H), \quad Z_1 = S_{5,7}^2(H), \quad Z_2 = S_{8,10}^2(H), \quad X = U[Z].$$

Теперь тензор X содержит для каждой пары смежных граней наборы векторов, ортогональных сторонам, длины которых равны длинам соответствующих сторон, направленных внутрь треугольной грани. Используя проекцию тензора H , мы вычислим два новых тензора XH^1, XH^2 , компоненты которых будут совпадать с векторами, ортогональными общей стороне каждой пары двух смежных граней (см. рис. 4).

$$X^1 = S_{1,1}^2(X), \quad X^2 = S_{2,2}^2(X),$$

$$XH^1 = \text{sum}_2(X^1 \cdot Z^1), \quad XH^2 = \text{sum}_2(X^2 \cdot Z^2).$$

Шаг 3. Теперь требуемый тензор получается вычислением $W = -(XH^1 + XH^2)$. Это равенство является следствием простого утверждения о том, что сумма внутренних нормалей треугольника, умноженных на длину соответствующей стороны, равна нулю. Теперь тензор G , содержащий градиент функции площади, может быть вычислен по формуле (6.1).

Введем в рассмотрение целочисленный вектор *boundary*. Его компоненты равны номерам тех вершин полигональной поверхности из матрицы P , которые мы задаем как граничные. Поскольку решение задачи Плато предполагает, что граница фиксирована, то в процессе алгоритма градиентного спуска это можно обеспечить задав в каждой граничной точке значение градиента площади равным нулю. Используя правила индексации (2.4), мы можем занулить градиент в каждой граничной

точке операций $G[\text{boundary}] = 0$. Отметим, что для заданной полигональной поверхности ее граничные вершины могут быть вычислены с помощью одного из алгоритмов, которые описаны в работе [2].

7. Тестирование

Минимальные поверхности являются экстремальными функционала площади. Поэтому в качестве первого этапа тестирования осуществлялась проверка градиента в точках минимума — на минимальных поверхностях. Для оценки погрешности мы использовали следующую норму градиента:

$$\|G\|_1 = \sum_{i=1}^N \sum_{k=1}^3 |G_{ik}|.$$

Тестирование проводилось на следующих поверхностях:

1. Плоскость $z = ax + by$.
2. Геликоид $z = \arctg \frac{y}{x}$.
3. Поверхность Шерка $z = \ln \cos x - \ln \cos y$.
4. Поверхность вращения, катеноид $x^2 + y^2 = \cosh z$.
5. Поверхность вращения, параболоид $z = x^2 + y^2$ — для сравнения с расчетными данными для минимальных поверхностей, являющихся экстремальными точками функционала площади. Результаты расчетов приведены в табл. 1. Вычисления проводились на триангуляции точек равномерной сетки 100×100 узлов в квадратной области.

Таблица 1. Проверка градиента на минимальных поверхностях

Поверхность	Норма градиента $\ G\ _1$
Плоскость	$5,983341426485378 \cdot 10^{-12}$
Геликоид	$2,198778895927487 \cdot 10^{-6}$
Поверхность Шерка	$1,011004799589836 \cdot 10^{-3}$
Катеноид	$9,899704132934952 \cdot 10^{-3}$
Параболоид	33,8248996514877

Малые значения нормы градиента соответствуют экстремальям функционала площади.

На следующем этапе тестирования проведены эксперименты по моделированию различных минимальных поверхностей. Рассмотрены следующие варианты:

Поверхность катеноида — минимальная поверхность вращения, задаваемая радиус-вектором

$$R(u, v) = \left(a \cosh \frac{u}{a} \cos v, a \cosh \frac{u}{a} \sin v, v \right), \quad 0 < v \leq 2\pi, -h < u < h.$$

Минимальная поверхность Шерка, которую можно задать параметрически следующим образом:

$$R(u, v) = \left(u, v, \log \frac{\cos u}{\cos v} \right), \quad -a \leq u, v \leq a, \quad a < \pi/2.$$

Наконец, рассмотрено моделирование поверхности геликоида

$$R(u, v) = (u \cos v, u \sin v, v), \quad 1 \leq u \leq 2, \quad 0 \leq v \leq \pi/2.$$

В качестве погрешности приближения рассмотрена абсолютная величина δS разности площадей расчетной поверхности и моделируемой поверхности. В табл. 2 приведены результаты численного эксперимента.

Таблица 2. Результаты эксперимента

Поверхность	Величина δS
Катеноид	$5,71 \cdot 10^{-4}$
Поверхность Шерка	$2,27 \cdot 10^{-3}$
Геликоид	$8,61 \cdot 10^{-6}$

Площади моделируемых полигональных поверхностей можно вычислить используя также тензорные операции (2.5), (2.6). Ниже приведены соответствующие формулы:

$$Q = P[T], \quad R = S_{2,3}^2(Q) - S_{1,1}^2(Q), \quad Z = \text{cross}(S_{1,1}^2(R), S_{2,2}^2(R)),$$

$$S = 0,5 \cdot \text{sum}_1 \left(\sqrt{\text{sum}_2(Z \cdot Z)} \right).$$

Приведем табл. 3 для сравнения с алгоритмом численного решения задачи Плато, построенного ранее в работе [2]. Показано время вычисления приближенной поверхности катеноида. В первом столбце указано время в секундах для алгоритма А, взятого из [2], а во втором столбце указано время алгоритма В, описанного в настоящей статье. Число итераций градиентного спуска в экспериментах было равно 1000.

Таблица 3. Время вычисления приближенной поверхности катеноида

Алгоритм А	Алгоритм В	Число вершин поверхности
18, 19	0, 23	121
90, 47	0, 53	441
339, 70	2, 06	1681

8. Заключение

Проведенные эксперименты показывают высокую эффективность использования тензорного представления расчетных схем. Сложность может заключаться в поиске такого рода представления. Алгоритмы вычисления полигональной поверхности, представленные выше, реализованы с использованием пакета NumPy для языка программирования Python. Исходный текст программ размещен в свободном доступе в репозитории автора статьи на ресурсе GitHub по адресу

<https://github.com/KlyachinVA/TensorOperations/tree/master>.

В дальнейшем предполагается применить приведенный подход для численного решения краевых задач нелинейной теории упругости. Обоснование существования точного решения таких задач дано в работах Дж. Болла [11], С. К. Водопьянова и А. О. Молчановой [1], а в работах В. В. Кузьмина и автора [4; 6] дано обоснование существования приближенного решения соответствующих вариационных задач и представлены алгоритмы их поиска.

Список источников

1. Водопьянов С. К., Молчанова А. О. Вариационные задачи нелинейной теории упругости в некоторых классах отображений с конечным искажением // Доклады Академии наук. 2015. Т. 465, № 5. С. 523–526. <https://doi.org/10.7868/S086956521535008X>
2. Григорьева Е. Г., Клячин А. А., Клячин В. А. Алгоритмы идентификации границы и визуализация решения задачи Плато в среде Blender // Научная визуализация. 2017. Т. 9, № 4. С. 13–25. <https://doi.org/10.26583/sv.9.4.02>
3. Дао Чонг Тхи, Фоменко А. Т. Минимальные поверхности и проблема Плато. М. : Наука, 1987. 312 с.
4. Клячин В. А., Кузьмин В. В. О существовании приближенных решений вариационных задач в нелинейной теории упругости // Известия Иркутского государственного университета. Серия Математика. 2025. Т. 53. С. 51–68. <https://doi.org/10.26516/1997-7670.2025.53.51>
5. Краснов М. М., Балашов В. А., Савенков Е. Б. Применение сеточно-операторного подхода для эффективной реализации явных разностных схем // Математическое моделирование. 2021. Т. 33, № 2. С. 20–40. <https://doi.org/10.20948/mm-2021-02-02>

6. Кузьмин В. В. Расчет 3D-формы гиперупругого тела для моделей нелинейной теории упругости методом Ньютона // Математическая физика и компьютерное моделирование. 2024. Т. 27, № 2. С. 80–91. <https://doi.org/10.15688/mpcm.jvolsu.2024.2.7>
7. О представлении разностных схем математической физики в операторной форме / А. А. Самарский, В. Ф. Тишкин, А. П. Фаворский, М. Ю. Шашков // Доклады АН СССР. 1981. Т. 258, № 5. С. 1092–1096.
8. Операторные разностные схемы / А. А. Самарский, В. Ф. Тишкин, А. П. Фаворский, М. Ю. Шашков // Дифференциальные уравнения. 1981. Т. 7, № 7. С. 1317–1327.
9. Саранин В. А. Равновесие жидкостей и его устойчивость. М. : Ин-т компьютер. исслед., 2002. 144 с.
10. Финн Р. Равновесные капиллярные поверхности. Математическая теория / с доб. Х. Уэнта ; пер. с англ. З. Я. Шапиро ; под ред. А. Т. Фоменко. М. : Мир, 1989. 310 с.
11. Ball J. M. Convexity conditions and existence theorems in nonlinear elasticity // Arch. Ration. Mech. Anal. 1977. N 63. P. 337–403.
12. Klyachin A. A., Klyachin V. A. Study of the existence and uniqueness of polygonal minimal surfaces // Lobachevskii Journal of Mathematics. 2025. Vol. 46, N 3. pp. 1139–1150. <https://doi.org/10.1134/S1995080225605156>

References

1. Vodop'yanov S.K., Molchanova A.O. Variational problems of nonlinear elasticity in certain classes of mappings with finite distortion. *Doklady Mathematics*, 2015, vol. 92, no. 3, pp. 739–742. <https://doi.org/10.7868/S086956521535008X> (in Russian)
2. Grigorieva E.G., Klyachin A.A., Klyachin V.A. Boundary identification algorithms and visualization of the solution of the Plateau problem in a Blender environment. *Scientific visualization*, 2017, vol. 9, no. 4, pp. 13–25. <https://doi.org/10.26583/sv.9.4.02> (in Russian)
3. Dao Trong Thi, Fomenko A.T. *Minimal Surfaces, Stratified Multivarifolds and the Plateau Problem*. Translation of Mathematical Monographs, 84, Amer. Math. Soc., 1991, 404 p.
4. Klyachin V.A., Kuzmin V.V. On the existence of approximate solutions of variational problems in nonlinear elasticity theory. *Bulletin of Irkutsk State University. Series Mathematics*, 2025, vol. 53, pp. 51–68. <https://doi.org/10.26516/1997-7670.2025.53.51> (in Russian)
5. Krasnov M.M., Balashov V.A., Savenkov E.B. A grid-operator framework for efficient implementation of explicit finite difference schemes. *Math. Models Comput. Simul.*, 2021, vol. 13, no. 5, pp. 831–843. <https://doi.org/10.20948/mm-2021-02-02> (in Russian)
6. Kuzmin V.V. Calculation of 3D Shape of a Hyperelastic Body for Nonlinear Elasticity Models Using the Newton Method. *Mathematical Physics and Computer Simulation*, 2024, vol. 27, no. 2, pp. 80–91. <https://doi.org/10.15688/mpcm.jvolsu.2024.2.7> (in Russian)
7. Samarskii A.A., Tishkin V.F., Favorskii A.P., Shashkov M.Y. On the representation of difference schemes of mathematical physics in operator form. *Soviet Physics Doklady*, 1981, vol. 26, pp. 1092–1096. (in Russian)

8. Samarskii A.A., Tishkin V.F., Favorskii A.P., Shashkov M.Y. Operator-difference schemes. *Differentsial'nye Uravneniya*, 1981, vol. 17, no. 7, pp. 1317–1327. (in Russian)
9. Saranin V.A. *Equilibrium of liquids and its stability*. Moscow, Institute of Computer Research, 2002. 144 p. (in Russian)
10. Finn R. *Equilibrium capillary surfaces*. Springer-Verlag, 1986, 245 p.
11. Ball J.M. Convexity conditions and existence theorems in nonlinear elasticity. *Arch. Ration. Mech. Anal.*, 1977, no. 63, pp. 337–403.
12. Klyachin A.A., Klyachin V.A. Study of the existence and uniqueness of polygonal minimal surfaces. *Lobachevskii Journal of Mathematics*, 2025, vol. 46, no. 3, pp. 1139–1150. <https://doi.org/10.1134/S1995080225605156>

Об авторах

Клячин Владимир

Александрович, д-р физ.-мат. наук, проф., Волгоградский государственный университет, Волгоград, 400062, Российская Федерация, klchnv@mail.ru, <https://orcid.org/0000-0003-1922-7849>

About the authors

Vladimir A. Klyachin, Dr. Sci.

(Phys.-Math.), Prof., Volgograd State University, Volgograd, 400062, Russian Federation, klchnv@mail.ru, <https://orcid.org/0000-0003-1922-7849>

Поступила в редакцию / Received 01.12.2025

Поступила после рецензирования / Revised 29.12.2025

Принята к публикации / Accepted 12.01.2026