



ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 193 за 2018 г.



ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Астафуров Г.О.

Алгоритм обхода ячеек в
характеристических методах
решения уравнения
переноса

Рекомендуемая форма библиографической ссылки: Астафуров Г.О. Алгоритм обхода ячеек в характеристических методах решения уравнения переноса // Препринты ИПМ им. М.В.Келдыша. 2018. № 193. 24 с. doi:[10.20948/prepr-2018-193](https://doi.org/10.20948/prepr-2018-193)
URL: <http://library.keldysh.ru/preprint.asp?id=2018-193>

**Ордена Ленина
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Российской академии наук**

Г.О.Астафуров

**Алгоритм обхода ячеек
в характеристических методах
решения уравнения переноса**

Москва — 2018

Астафуров Г.О.

Алгоритм обхода ячеек в характеристических методах решения уравнения переноса

При решении уравнения переноса интерполяционно-характеристическим методом на неструктурированной сетке возникает задача экономичного алгоритма обхода разрешимых ячеек. В предлагаемой работе описана реализация этого алгоритма, основанная на топологической сортировке ориентированного графа. В силу этого предлагаемый алгоритм является наиболее экономичным из всех возможных. Приведена параллельная реализация алгоритма обхода в случае, когда расчетная сетка разбита на отдельные области, хранимые на разных узлах вычислительного кластера. Существуют сетки, для которых интерполяционно-характеристический метод приводит к неразрешимой задаче. Метод позволяет осуществлять диагностику таких сеток.

Ключевые слова: уравнение переноса, интерполяционно-характеристический метод, обход ячеек, топологическая сортировка

Gleb Olegovich Astafurov

Cells' bypassing algorithm for characteristic methods of transport equation solving

When solving the transport equation by the interpolation-characteristic method on an unstructured grid, the problem of an economical algorithm for queuing solvable cells arises. This paper describes the implementation of this algorithm based on the topological sorting of the oriented graph. Therefore, the proposed algorithm is the most economical of all possible ones. A parallel implementation of the bypass algorithm is presented in the case when the computational grid is divided into separate areas stored on different nodes of the computational cluster. There are grids for which the interpolation-characteristic method leads to an unsolvable problem. Proposed method makes it possible to diagnose such grids.

Key words: transport equation, interpolation-characteristic method, cells' bypassing, topological sorting

Работа выполнена при поддержке Российского фонда фундаментальных исследований, проект 18-01-00857.

Оглавление

Введение.....	3
Неструктурированная сетка	6
Сеточный граф.....	10
Последовательный алгоритм обхода ячеек	12
Параллельный алгоритм обхода ячеек в случае Domain Decomposition	14
Применимость интерполяционно-характеристического метода.....	21
Заключение	23
Библиографический список.....	23

Введение

Необходимость численного решения уравнения переноса неполяризованного излучения или незаряженных частиц появляется во многих задачах науки и техники. К таким задачам относится расчет переноса собственного излучения плазмы или расчет переноса нейтронов в задачах защиты или при моделировании активных зон реакторов. Уравнение переноса является интегро-дифференциальным. Дифференциальная часть этого уравнения представляется простейшим уравнением в частных производных первого порядка, а для учета интегрального члена рассеяния используются те или иные методы итерационного типа. В данном препринте мы не занимаемся вопросами, связанными с рассеянием, поэтому считаем правую часть уравнения известной.

Рассмотрим простейший случай стационарного уравнения переноса излучения без рассеяния [1]

$$\Omega_1 \frac{\partial I}{\partial x_1} + \Omega_2 \frac{\partial I}{\partial x_2} + \Omega_3 \frac{\partial I}{\partial x_3} + \sigma I = \sigma I_p. \quad (1)$$

Здесь $I = I(x, \Omega)$ – интенсивность излучения в точке $x = (x_1, x_2, x_3) \in D$ расчетной области $D \subset \mathbb{R}^3$ и в направлении $\Omega = (\Omega_1, \Omega_2, \Omega_3)$, $|\Omega| = 1$. Функции $I_p = I_p(x)$ и $\sigma = \sigma(x)$ известны и представляют собой соответственно интенсивность равновесного излучения источников и коэффициент поглощения излучения в точке $x \in D$.

Важным свойством уравнения (1) является наличие характеристик. Вдоль прямой $x_0 + s\Omega$ уравнение (1) для функции $I(s) = I(x_0 + s\Omega, \Omega)$ сводится к обыкновенному дифференциальному уравнению

$$\frac{dI(s)}{ds} + \sigma(s)I(s) = \sigma(s)I_p(s),$$

точное решение которого находится по формуле

$$I(s) = \left(I(0) + \int_0^s \frac{\sigma(\tau)I_p(\tau)}{\exp\left(-\int_0^\tau \sigma(\xi)d\xi\right)} d\tau \right) \exp\left(-\int_0^s \sigma(\tau)d\tau\right). \quad (2)$$

Формула (2) выражает значение интенсивности в точке $x_0 + s\Omega$ через ее значение в точке x_0 . В случае, когда функции $\sigma = \sigma(s) = \sigma(x_0 + s\Omega)$ и

$I_p = I_p(s) = I_p(x_0 + s\Omega)$ являются константами, формула (2) значительно упрощается:

$$I(s) = (I(0) - I_p) \exp(-\sigma s) + I_p. \quad (3)$$

В соответствии с характеристическим свойством уравнение (1) должно быть дополнено краевыми условиями. В качестве x_0 можно взять точку на освещенной границе области D , в которой вектор внешней нормали n_{x_0} образует тупой угол с направлением Ω . Для нахождения решения уравнения (1) необходимо поставить граничные условия

$$I(x, \Omega) = I_0(x, \Omega), \text{ для всех } x \in \partial D, \langle \Omega, n_x \rangle < 0. \quad (4)$$

Задача переноса излучения (1), (4) является корректно поставленной.

На характере поведения решения уравнения (1) вдоль характеристик основан ряд численных методов. К методам этого класса относится интерполяционно-характеристический метод. В этом методе на освещенной границе ячейки решение интерполируется тем или иным образом, а потом интегрируется вдоль характеристик. Если структура сетки простая (например, состоит из прямоугольных ячеек), то обычно разрешение ячеек производится методом бегущего счета одна за другой. Ситуация радикально меняется при использовании неструктурированных сеток, например из тетраэдров. Такие сетки являются более адекватными для описания сложных геометрических структур и потому широко распространены в задачах техники. В этом случае при решении задачи переноса возникают две подзадачи:

- нахождение решения в ячейке-тетраэдре по той или иной схеме, базирующейся на характеристических свойствах уравнения;
- нахождение порядка разрешения ячеек.

В данном препринте мы будем апеллировать к численной схеме, изложенной в работе [2]. Эта схема основана на интерполяции второго порядка в барицентрических координатах. Однако актуальность экономичного алгоритма обхода ячеек значительно шире, и предлагаемый в данном препринте алгоритм может быть применен, к примеру, к схемам [3] и [4].

Для того чтобы воспользоваться формулой (3) при численном решении задачи переноса, требуется достаточно мелкая пространственная сетка, в каждой ячейке которой величины σ и I_p можно считать примерно постоянными. Переход от (2) к (3) при постоянных величинах σ и I_p обеспечивает не более чем второй порядок аппроксимации.

Сеточные величины при построении численной схемы задаются на элементах сетки. Например, в базовой схеме [2] интенсивность излучения I задавалась в узлах, отрезках и треугольниках сетки, а величины σ и I_p задавались в ячейках и считались постоянными. Значение интенсивности

излучения на отрезке или на треугольнике интерпретировалось как интегральное среднее интенсивности по этому отрезку или треугольнику. Схема [2] имеет второй порядок сходимости к гладкому решению и консервативна.

Ячейка сетки называется разрешимой, если известны все сеточные величины, входящие в ее освещенные двумерные грани, и разрешенной, когда известны все сеточные величины, связанные с этой ячейкой. Имеются три типа ячеек в зависимости от количества освещенных граней–треугольников.

Как уже было сказано, интерполяционно-характеристический метод заключается в интерполяции интенсивности излучения на освещенных гранях разрешимой ячейки (при помощи известных на них сеточных величин) и использовании ее для вычисления недостающих сеточных величин на неосвещенных гранях при помощи формулы (3). В начале расчета сеточная интенсивность известна из граничного условия (4) лишь на освещенных элементах границы расчетной сетки. Разрешение ряда начальных ячеек меняет границу разрешимости. Граница разрешимости ячеек является динамической характеристикой процесса. В силу этого экономичный алгоритм правильного обхода ячеек сетки является одним из основных структурных элементов численного метода. Простейший алгоритм перебора для поиска разрешимой ячейки является совершенно неудовлетворительным с точки зрения количества затрачиваемых на поиск ресурсов.

Данная работа посвящена созданию алгоритма обхода ячеек на основе теории графов. Можно свести поставленную задачу к задаче топологической сортировки (упорядочению вершин, в соответствии с направлением стрелок) специального ориентированного графа, связанного с сеткой. Как известно, возможность топологической сортировки эквивалентна отсутствию ориентированных циклов в графе. Предлагаемый алгоритм также способен дать ответ на вопрос о применимости характеристического метода к данной сетке.

В радиационной газовой динамике [1] важную роль играют такие величины, как плотность и поток энергии излучения

$$U(x) = \int_{4\pi} I(x, \Omega) d\Omega, \quad W(x) = \int_{4\pi} I(x, \Omega) \Omega d\Omega. \quad (5)$$

Для их нахождения задача (1), (4) должна быть решена для некоторого набора угловых направлений Ω , определяющих выбранную квадратурную формулу.

В случае наличия граничных условий отражения вместо условий (4) на границе расчетной области ставятся более сложные условия зеркально-диффузного отражения

$$I(x, \Omega) = (1 - r_{Diff} - r_{Spec}) I_0(x, \Omega) + r_{Diff} I_{Diff}(x) + r_{Spec} I_{Spec}(x, \Omega), \quad (6)$$

для всех $x \in \partial D$, $\langle \Omega, n_x \rangle < 0$.

Здесь $I_{Diff}(x)$ – интенсивность диффузно отраженного излучения в точке $x \in \partial D$, получаемая из соотношения

$$I_{Diff}(x) \cdot \int_{\langle \Omega, n_x \rangle \leq 0} \langle \Omega, n_x \rangle d\Omega + \int_{\langle \Omega, n_x \rangle \geq 0} I(x, \Omega) \langle \Omega, n_x \rangle d\Omega = 0.$$

А $I_{Spec}(x, \Omega)$ – интенсивность зеркально отраженного излучения в точке $x \in \partial D$ в направлении Ω , получаемая из соотношения

$$I_{Spec}(x, \Omega) = I(x, \Omega - 2\langle \Omega, n_x \rangle n_x).$$

Здесь $r_{Diff} = r_{Diff}(x)$ и $r_{Spec} = r_{Spec}(x)$ – известные коэффициенты диффузного и зеркального отражения в точке $x \in \partial D$, для которых выполнено неравенство

$$r_{Diff}(x) + r_{Spec}(x) < 1.$$

Для численного решения задачи переноса с отражением (1), (6) а также для вычисления интегральных величин (5) прибегают к использованию сферических кубатурных формул [5], [6]: дискретных аналогов интегрирования по двумерной сфере единичного радиуса

$$\int_{4\pi} f(\Omega) d\Omega \approx \sum_{i=1}^N w_i f(\Omega_i). \quad (7)$$

Направления $\Omega_1, \dots, \Omega_N$ называются узлами или углами кубатурной формулы (1.7), а числа $w_1, \dots, w_N > 0$ – ее весами.

Задача (1), (6) решается методом простых итераций. Сначала для всех угловых направлений решается задача (1), (4) без учета отражения. Затем из попавшего на границу излучения рассчитывается отраженное излучение, граничные условия пересчитываются, и снова решается задача вида (1), (4). И так до тех пор, пока не будет достигнута требуемая точность.

В этой работе мы не будем детально заниматься вопросами построения кубатурных формул и реализацией граничных условий зеркально-диффузного отражения. Однако при построении алгоритмов разрешения ячеек и структур данных для хранения элементов сетки и сеточных величин мы будем держать прицел и на эти задачи тоже.

Неструктурированная сетка

Прежде чем приступить к решению задачи упорядочения ячеек в интерполяционно-характеристическом методе, нам понадобится строгое определение неструктурированной сетки из тетраэдров, которое впоследствии поможет нам построить удобную структуру данных для хранения сетки в памяти компьютера.

Такая сетка задается набором (S_0, S_1, S_2, S_3) из четырех конечных множеств симплексов (S – simplex) размерностей ноль, один, два и три соответственно. А именно: множеством узлов сетки S_0 ; множеством отрезков S_1 , концами которых являются узлы из множества S_0 ; множеством S_2 треугольников, стороны которых являются отрезками из множества S_1 , и множеством S_3 ячеек-тетраэдров, имеющих треугольные грани из множества S_2 .

Тетраэдр инцидентен одной трехмерной грани (самому себе), четырем двумерным граням – треугольникам, шести одномерным граням – отрезкам и четырем нульмерным граням – узлам. Построенные по аналогии правила инцидентности имеют место для треугольников, отрезков и узлов.

Потребуем, чтобы любая грань симплекса из множества $S = S_0 \cup S_1 \cup S_2 \cup S_3$ являлась элементом S . Также потребуем, чтобы пересечение любых двух симплексов из множества S либо было пусто, либо являлось общей гранью обоих симплексов.

Удовлетворяющий перечисленным требованиям набор (S_0, S_1, S_2, S_3) называется трехмерным симплициальным комплексом. Действуя аналогично, можно определить симплициальные комплексы любой размерности, в частности одномерные и двумерные, с которыми мы также будем иметь дело.

Ясно, что для определения неструктурированной сетки из тетраэдров понятие трехмерного симплициального комплекса является слишком общим. Сузим его, определив дополнительные качества таких комплексов.

Трехмерный комплекс (S_0, S_1, S_2, S_3) однороден, если любой его симплекс инцидентен некоторому трехмерному симплексу. То есть в однородном комплексе каждый элемент множества $S = S_0 \cup S_1 \cup S_2 \cup S_3$ входит в некоторый тетраэдр из множества S_3 .

Трехмерный комплекс (S_0, S_1, S_2, S_3) неразветвлен, если каждый двумерный симплекс инцидентен не более чем двум симплексам размерности три. То есть в неразветвленном комплексе каждый треугольник из множества S_2 входит не более чем в два тетраэдра из множества S_3 .

В однородном неразветвленном трехмерном комплексе все треугольники, входящие ровно в один тетраэдр, порождают двумерный симплициальный комплекс (BS_0, BS_1, BS_2) , называемый границей или краем исходного комплекса (BS – boundary simplex). Если края нет, то комплекс называется замкнутым.

Трехмерный комплекс (S_0, S_1, S_2, S_3) называется сильно связанным, если любые два его тетраэдра можно соединить цепочкой тетраэдров, в которой соседи граничат по треугольнику. То есть в сильно связанном комплексе для любых двух элементов T, T' множества S_3 найдется последовательность $T_1, \dots, T_n \in S_3$ тетраэдров, в которой $T_1 = T$, $T_n = T'$ и любые два соседних элемента T_i, T_{i+1} имеют общую двумерную грань $F_i \in S_2$.

Однородный неразветвленный сильно связанный трехмерный комплекс называется трехмерным псевдомногообразием.

Для строгого определения неструктурированных сеток из тетраэдров понятие трехмерного псевдомногообразия все еще довольно широко, и требуется более специальное понятие трехмерного ориентируемого многообразия [7]. Однако для решения нашей задачи псевдомногообразия уже достаточно, поэтому остановимся на нем.

Заметим, что мы также неявно подразумеваем вложенность нашей сетки в пространство \mathbb{R}^3 , что также является сильным ограничением на общее понятие симплициального комплекса.

При решении задач обтекания расчетные области топологически являются трехмерными областями с пустотами – только такие конфигурации нас интересуют. Заметим, что граничные условия (4) или (6) не подразумевают перенос излучения через пустоты. Поэтому дополнительно потребуем, чтобы граница нашей сетки являлась объединением двумерных замкнутых псевдомногообразий. Однородность для них понимается как инцидентность любого граничного симплекса некоторому граничному треугольнику; неразветвленность и замкнутость – как инцидентность любого граничного отрезка ровно двум граничным треугольникам. Заявленными двумерными замкнутыми псевдомногообразиями будут компоненты сильной связности граничного двумерного комплекса (BS_0, BS_1, BS_2) .

Итак, неструктурированной сеткой называется вложенное в пространство \mathbb{R}^3 трехмерное псевдомногообразие (S_0, S_1, S_2, S_3) , граница (BS_0, BS_1, BS_2) которого распадается в объединение двумерных замкнутых псевдомногообразий.

Структура данных, задающая сетку (S_0, S_1, S_2, S_3) в компьютере, должна быть экономной с точки зрения занимаемой памяти и с точки зрения скорости исполнения алгоритмов, работающих с ней. Баланс между этими двумя требованиями зависит от характера задач, поставленных перед сеткой. Наиболее распространенная реализация сетки [8] в объектно-ориентированных языках программирования строится так:

- объект «узел» содержит три поля для пространственных координат;
- объект «отрезок» имеет два поля для указателей на узлы, являющиеся его концами;
- объект «треугольник» имеет три поля для указателей на отрезки и два поля для указателей на ячейки (в силу неразветвленности), которые инцидентны ему;
- а объект «ячейка» имеет четыре поля для указателей на инцидентные ей треугольники.

Если некоторый треугольник инцидентен лишь одной ячейке, то оставшееся поле под указатель на ячейку занимает нулевой указатель `NULL`. Все узлы,

отрезки, треугольники и ячейки сетки объединяются в списки. Эти четыре списка полностью задают сетку в памяти компьютера.

Такая структура данных экономна по памяти и позволяет быстро переходить из ячейки в соседние ячейки: сначала из данной ячейки осуществляется переход по указателю в инцидентную ей двумерную грань, затем из этой грани переход по указателю в инцидентную ей ячейку, отличную от изначальной. Переходы между соседними ячейками понадобятся нам в большом количестве при реализации алгоритма обхода.

Для работы с рассчитываемыми сеточными величинами в определении объектов добавляются поля под эти величины. В данной работе все программы были написаны на языке C++.

В задачах переноса излучения (1), (4) или (1), (6) возникает необходимость решать уравнение переноса (1) для различных угловых направлений $\Omega_1, \dots, \Omega_N$ из кубатурной формулы (7). В последовательной реализации алгоритма разрешения нецелесообразно хранить в элементах сетки сеточную интенсивность во всех угловых направлениях. Это будет занимать лишнюю память. Достаточно в определении каждого элемента сетки иметь одно поле под значение интенсивности излучения в текущем угловом направлении и два поля под рассчитываемые величины плотности и потока энергии излучения (5). В ходе расчета в очередном угловом направлении в соответствии с формулой (7) вычисляются вклады интенсивностей в интегральные величины (5). Однако при реализации граничных условий (4) или (6) в каждом граничном узле, отрезке и треугольнике следует хранить полный список интенсивностей излучения для всех узлов $\Omega_1, \dots, \Omega_N$ кубатурной формулы. Таким образом, для работы с границей (BS_0, BS_1, BS_2) расчетной сетки (S_0, S_1, S_2, S_3) необходимо задать отдельную структуру данных.

Приведем определения объектов в структуре данных границы сетки по аналогии с тем, как это делалось выше в структуре данных самой расчетной сетки, опуская упоминание полей под списки сеточных интенсивностей.

- Объект **«граничный узел»** имеет одно поле под указатель на соответствующий узел (из структуры данных для сетки).
- Объект **«граничный отрезок»** имеет одно поле под указатель на соответствующий отрезок, два поля под указатели на граничные узлы, являющиеся его концами, и два поля под указатели на граничные треугольники, которые ему инцидентны.
- Объект **«граничный треугольник»** имеет одно поле под указатель на соответствующий треугольник и три поля под указатели на граничные отрезки, инцидентные ему.

Все граничные узлы, отрезки и треугольники объединяются в списки. Эти три списка задают границу сетки в памяти компьютера.

Сеточный граф

Пусть в пространстве \mathbb{R}^3 задана неструктурированная сетка из тетраэдров (S_0, S_1, S_2, S_3) . Введем вспомогательный объект – неориентированный граф $G=(V, E)$, вершины которого составляют множество $V = S_3$ всех ячеек сетки, а ребра составляют множество $E \subset S_2$ всех двумерных граней, не лежащих на границе сетки. Будем считать, что две вершины $v_1, v_2 \in V$ соединены ребром $e \in E$ в том и только том случае, если соответствующие тетраэдры v_1, v_2 имеют общий треугольник e . Можно считать, что доступ к сеточному графу G в вышеопределенной структуре данных организован через списки смежности вершин графа.

Подчеркнем, что в нашей работе узлы сетки не называются вершинами и отрезки сетки не называются ребрами, а вершинами и ребрами называются только вершины и ребра графа. Это сделано во избежание путаницы.

Вернемся к задаче переноса излучения (1), (4) на неструктурированной сетке. Для фиксированного направления излучения Ω задается ориентация ребер графа G . Ребро $e \in E$ получает направление из вершины $v_1 \in V$ в вершину $v_2 \in V$ тогда и только тогда, когда грань e освещена в ячейке v_2 и не освещена в ячейке v_1 . То есть если внешняя нормаль ячейки v_1 (внутренняя нормаль ячейки v_2), исходящая из грани e , образует острый угол с направлением Ω . В случае, когда нормаль грани e ортогональна Ω , ориентация ребра e в графе G может быть задана произвольно при условии, что такой выбор направления не будет замыкать стрелки в цикл (об этом подробнее пойдет речь дальше). Множество стрелок, полученных из ребер E , обозначается через E_Ω . А ориентированный граф (V, E_Ω) обозначается G_Ω .

На рис. 1 изображены вершины графа G_Ω , соответствующие внутренним ячейкам первого, второго и третьего типа освещенности [2]. Напомним, что тип освещенности ячейки определяется количеством освещенных двумерных граней. У граничных ячеек будет нехватать каких-то входящих или исходящих стрелок. Количество входящих в вершину стрелок называется ее входящей степенью. Тип освещенности можно отождествлять с входящей степенью соответствующей вершины графа G_Ω только для внутренних ячеек.

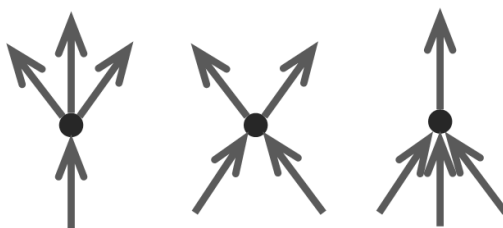


Рис. 1. Ячейки первого, второго и третьего типа

Отметим несколько связей между графом G_Ω и порядком разрешения ячеек в интерполяционно-характеристическом методе.

- Если стрелка $e \in E_\Omega$ направлена из вершины $v_1 \in V$ в вершину $v_2 \in V$, то разрешение ячейки v_1 предшествует разрешению ячейки v_2 .
- Предположим, что в графе G_Ω имеется цикл, то есть существует последовательность вершин $v_1, \dots, v_n \in V$, в которой $v_1 = v_n$ и соседние вершины v_i, v_{i+1} соединены стрелкой. Тогда интерполяционно-характеристический метод неприменим для решения уравнения переноса излучения (1) в направлении Ω .

Действительно, в противном случае ячейка v_1 разрешалась бы прежде, чем ячейка v_2 , v_2 – прежде v_3 , и так далее, v_{n-1} – прежде v_n , но $v_1 = v_n$ – противоречие.

- Если граф G_Ω не содержит циклов, то решение уравнения переноса (1) в направлении Ω интерполяционно-характеристическим методом возможно.

Это утверждение основывается на известном факте [9] из теории графов, гласящем, что вершины ориентированного ациклического графа можно топологически упорядочить. То есть так пронумеровать все вершины $v_1, \dots, v_{|V|} \in V$, что наличие стрелки $e \in E_\Omega$ из вершины v_i в вершину v_j влекло бы неравенство $i < j$. Такая нумерация вершин графа G_Ω задает правильный порядок разрешения ячеек в интерполяционно-характеристическом методе.

Таким образом, имеет место

Утверждение 1. Применимость интерполяционно-характеристического метода к решению уравнения переноса излучения (1) в направлении Ω равносильна ациклическости графа G_Ω .

Нужно как-то отобразить в структуре данных неструктурированной сетки (S_0, S_1, S_2, S_3) направления стрелок в графе G_Ω . Для этого в определении объектов треугольников будем учитывать порядок следования полей для указателей на инцидентные ячейки. А именно, будем считать, что в треугольнике $t \in S_2$ указатели на инцидентные ему ячейки $c, c' \in S_3 \cup \{\text{NULL}\}$ идут в порядке $c \rightarrow c'$ (указатель NULL означает, что треугольник t расположен на границе сетки и инцидентен лишь одной ячейке), и писать $t(c \rightarrow c')$, если и только если грань t не освещена в ячейке c или освещена в ячейке c' . Дизъюнкция здесь позволяет учесть случай, когда c либо c' не принадлежит S_3 . Если на первом месте окажется NULL, то треугольник t

лежит на освещенной границе сетки, а если на втором – то на неосвещенной границе.

Последовательный алгоритм обхода ячеек

Алгоритм основан на следующем свойстве ациклических графов

Утверждение 2. В ациклическом ориентированном графе имеется хотя бы один исток – вершина нулевой входящей степени (в которую не входят стрелки).

Действительно, в противном случае, постоянно переходя по стрелкам назад, мы рано или поздно замкнулись бы в цикл, в силу конечности количества вершин.

После удаления истока из графа вместе со всеми исходящими из него стрелками ацикличность графа сохраняется. Значит, можно снова найти исток. И так далее, пока не исчерпаются все вершины. Порядок последовательного удаления истоков совпадает с порядком вершин при топологической сортировке.

Применительно к графу G_{Ω} заметим, что на любой стадии процесса каждый удаляемый исток соответствует разрешимой ячейке. В согласии с этим принципом разрешение ячеек осуществляется последовательно по всем угловым направлениям, и вносимый вклад от каждого направления учитывается в ходе вычисления интегральных величин (5) по кубатурной формуле (7).

Алгоритм 1

Вход: Заданы неструктурированная сетка (S_0, S_1, S_2, S_3) с границей (BS_0, BS_1, BS_2) . Задана кубатурная формула (7) с узлами $\Omega_1, \dots, \Omega_N$. Поставлены граничные условия (4): для всех угловых направлений задана интенсивность в сеточных элементах освещенной границы.

Выход: Получено решение задачи переноса (1), (4) интерполяционно-характеристическим методом в соответствии с правильным порядком обхода ячеек: интегральные величины (5) рассчитаны во всех элементах сетки (S_0, S_1, S_2, S_3) , интенсивности излучения вычислены во всех элементах границы (BS_0, BS_1, BS_2) и во всех направлениях $\Omega_1, \dots, \Omega_N$. Либо получен сигнал о наличии цикла в графе G_{Ω_i} при некотором значении i .

(Ниже представлен псевдокод алгоритма разрешения ячеек)

Для всех i от 1 до N :

Во всех элементах сетки (S_0, S_1, S_2, S_3) очистить поле под значение интенсивности излучения в текущем угловом направлении;

Для всех треугольников $t(c \rightarrow c') \in S_2$:

Вычислить вектор нормали n к треугольнику t , направленный из ячейки c или в ячейку c' ;

Если $\langle n, \Omega_i \rangle \leq 0$:

В объекте t поменять местами указатели на ячейки c и c' ;

Инициализировать массив deg входящих степеней ячеек;

Для всех ячеек $c \in S_3$:

Присвоить $\text{deg}[c] := 0$;

Для всех треугольников $t(c' \rightarrow c'')$, инцидентных ячейке c :

Если $c'' = c$:

Увеличить $\text{deg}[c]$ на единицу;

Для всех граничных треугольников $bt \in BS_2$:

Рассмотреть соответствующий bt треугольник $t(c \rightarrow c') \in S_2$;

Если $c = \text{NULL}$:

Записать только оригинальные данные (повторно полученные данные не перезаписывать) из элементов граничного треугольника bt для направления Ω_i в элементы треугольника t ;

Учесть вклад от оригинальных данных в расчет интегральных величин (5) в этих элементах;

Уменьшить $\text{deg}[c']$ на единицу;

Инициализировать очередь Q для указателей на разрешимые ячейки;

Для всех ячеек $c \in S_3$:

Если $\text{deg}[c] = 0$:

Поместить указатель на c в конец очереди Q ;

Инициализировать счетчик $K := 0$ для разрешенных ячеек;

Пока очередь Q не пуста:

Достать ячейку c из начала очереди Q ;

Разрешить ячейку c ;

Записать только оригинальные данные (повторно полученные данные не перезаписывать) в элементы неосвещенных двумерных граней ячейки c ;

Учесть вклад от оригинальных данных в расчет интегральных величин (5) в этих элементах;

Увеличить K на единицу;

Для всех треугольников $t(c' \rightarrow c'')$, инцидентных ячейке c :

Если $c' = c$ и $c'' \neq \text{NULL}$:

Уменьшить $\text{deg}[c'']$ на единицу;

Если $\text{deg}[c''] = 0$:

Поместить указатель на c'' в конец очереди Q ;

Если $c'' = \text{NULL}$:

Произвести поиск граничного треугольника $bt \in BS_2$,
которому соответствует треугольник t ;

Записать данные из элементов треугольника t в элементы
граничного треугольника bt для направления Ω_i ;

Если $K \neq |S_3|$:

Сообщить, что имеется цикл в графе G_{Ω_i} ;

Конец алгоритма;

В алгоритме используется очередь. Это распространенный контейнер объектов, поддерживающий две операции: запись объекта в конец очереди и считывание объекта с последующим удалением его из начала очереди. Требование не учитывать повторно рассчитанную интенсивность в одном и том же элементе сетки необходимо для правильного учета вклада этой величины в расчет плотности и потока (5) в данном элементе.

Параллельный алгоритм обхода ячеек в случае Domain Decomposition

Как правило, в реальных задачах неструктурированная сетка является столь подробной, что объем связанных с ней данных не удастся разместить в памяти одного компьютера. Задача может быть решена при помощи суперкомпьютера. Прибегают к разбиению расчетной области на несколько подобластей (Domain Decomposition). Каждая часть расположена в памяти и обрабатывается одним вычислительным узлом кластера. Задача разрешения ячеек распадается на подзадачи для частей. Необходимость передавать данные между подзадачами возникает только при переходе через общую границу соответствующих подобластей. Программный интерфейс MPI [10], используемый для систем с распределенной памятью, реализует передачу сообщений между узлами кластера. Цикл по всем кубатурным направлениям присутствует в каждой MPI-подзадаче. Вычислительные узлы являются многоядерными процессорами, благодаря чему возможно локальное распараллеливание подзадач по кубатурным углам для ускорения работы метода. На втором уровне параллельности используется директивная технология OpenMP [11] для систем с общей памятью. Внешний обмен

сообщениями происходит между внутренними нитями MPI-процессов с одинаковыми угловыми направлениями.

Суть метода заключается в том, что на каждом MPI-узле для каждого углового направления создается нить, решающая задачу переноса излучения в данной части сетки в данном направлении. Выполнение нитей распараллеливается по ядрам MPI-процессора с помощью технологии OpenMP. Прежде всего, для всех нитей задаются граничные условия на освещенных частях внешних границ подобластей сетки. Однако в отличие от последовательной программы, граничные условия оказываются не определенными на освещенных частях смежных границ. Поэтому MPI-подзадачи не удастся разрешить независимо. А недостающие граничные условия появляются в ходе расчета и передаются в виде сообщений в нужные нити.

Алгоритм обхода ячеек для конкретной нити претерпевает незначительные изменения. Теперь имеется два режима работы алгоритма обхода:

- **Режим разрешения.** Ничем не отличается от разрешения ячеек в последовательном алгоритме. Запускается, если очередь разрешимых ячеек данной нити не пуста. При разрешении ячейки, неосвещенный треугольник которой является смежным с другим процессом, осуществляется передача данных с этого треугольника в нить смежного процесса, соответствующую тому же угловому направлению.
- **Режим ожидания.** Запускается, если очередь разрешимых ячеек данной нити оказывается пустой. Происходит прием сообщений из нитей смежных процессов с тем же угловым направлением. Принятые данные записываются в локальную память нити, понижается входящая степень соответствующей ячейки. Если входящая степень этой ячейки становится равной нулю, то указатель на нее добавляется в очередь разрешимых ячеек и осуществляется переход в первый режим.

Когда очереди разрешимых ячеек окажутся пустыми для нитей всех MPI-процессов какого-то углового направления, необходимо сравнить общее количество ячеек в процессах с количеством ячеек, разрешенных на этих нитях в ходе расчета. Если везде окажется равенство, то метод корректно отработал в этом угловом направлении. Противный случай свидетельствует о наличии цикла в ориентированном сеточном графе и неприменимости интерполяционно-характеристического метода. Контроль работы MPI-процессов осуществляет специальный корневой процесс, который в каждый момент времени имеет информацию о текущем режиме работы каждой нити каждого процесса. Корневой процесс прерывает работающие нити, если все очереди окажутся пустыми. Необходимо потребовать, чтобы порядок подачи нитей на ядра в циклах по углам был одинаковым на всех MPI-процессорах. Этого можно добиться средствами OpenMP.

Обсудим структуры данных, задающие расчетную сетку (S_0, S_1, S_2, S_3) , распределенную по узлам вычислительного кластера. Пусть множество ячеек S_3 разбито на P дизъюнктивных сильно связанных подмножеств S_3^1, \dots, S_3^P . Все грани меньших размерностей, инцидентные тетраэдрам из S_3^i , задают сетки $(S_0^i, S_1^i, S_2^i, S_3^i)$ для подобластей.

Заметим, что каждый треугольник $t \in S_2$ может входить не более чем в две подобласти $(S_0^i, S_1^i, S_2^i, S_3^i)$. Аналогичное утверждение неверно для отрезков и узлов из S_1 и S_0 . Поэтому при разрешении ячеек, неосвещенный треугольник которых расположен на стыке двух процессов, осуществляется передача в смежный процесс сеточных интенсивностей со всех элементов этого треугольника. В связи с этим замечанием передача данных стандартизирована под формат двумерных граней. Из полученных данных учитываются только оригинальные для принимающего процесса. При таком подходе вычисления интегральных величин (5) будут выполняться правильно.

Из сказанного выше следует, что для задания подобластей $(S_0^i, S_1^i, S_2^i, S_3^i)$ в памяти MPI-процессов годится та же структура данных, что и для последовательного алгоритма, с тем лишь отличием, что в определениях элементов сетки не будет поля под интенсивность излучения в текущем угловом направлении. Это связано с тем, что данные о сетке $(S_0^i, S_1^i, S_2^i, S_3^i)$ являются общими для всех OpenMP-нитей этого процесса, в то время как наборы интенсивностей для узлов, отрезков и треугольников в текущем угловом направлении являются локальными для каждой нити. Данные об интегральных величинах (5) по-прежнему привязаны к элементам сетки и являются общими. Также в локальной памяти каждой нити задается ориентация стрелок сеточного графа подобласти для соответствующего кубатурного направления. Обозначим локальные массивы сеточных интенсивностей Ω_s -ой нити i -го MPI-узла через $I_0^{i,s}$, $I_1^{i,s}$, $I_2^{i,s}$ соответственно для узлов отрезков и треугольников подобласти $(S_0^i, S_1^i, S_2^i, S_3^i)$. Ориентация стрелок сеточного графа $G_{\Omega_s}^i$ подобласти $(S_0^i, S_1^i, S_2^i, S_3^i)$ задается массивом битов $\text{ord}^{i,s}$ так, что $\text{ord}^{i,s}[t] = 1$, если в треугольнике $t(c \rightarrow c') \in S_2^i$ порядок указателей на ячейки $c \rightarrow c'$ согласован с направлением Ω_s , и $\text{ord}^{i,s}[t] = 0$ в противном случае.

Граница (BS_0^i, BS_1^i, BS_2^i) подобласти $(S_0^i, S_1^i, S_2^i, S_3^i)$ разбивается на две части $(CBS_0^i, CBS_1^i, CBS_2^i)$ и $(OBS_0^i, OBS_1^i, OBS_2^i)$: смежную границу с другими подобластями и внешнюю границу соответственно (CBS – common boundary simplex, OBS – outer boundary simplex). Каждый тип границы будет описываться

своей структурой данных, которая будет храниться в памяти соответствующего MPI-процесса и являться общей для всех локальных нитей.

Общая граница $(CBS_0^i, CBS_1^i, CBS_2^i)$ описывается одним лишь списком общих граничных треугольников.

- Объект «**общий граничный треугольник**» имеет одно поле под указатель на соответствующий треугольник из S_2^i . Три поля под указатели на инцидентные узлы из S_0^i и три поля под указатели на инцидентные отрезки из S_1^i . Восемь полей под целые числа: номер j смежного процесса, номер рассматриваемого треугольника в списке S_2^j смежного процесса, а также номера инцидентных узлов из S_0^j и отрезков из S_1^j , записанные в правильном порядке.

Передаваемое сообщение через общий граничный треугольник имеет вид $[T, IT, L_1, IL_1, L_2, IL_2, L_3, IL_3, P_1, IP_1, P_2, IP_2, P_3, IP_3]$, где T – номер этого треугольника в принимающем процессе, IT – интенсивность в нем, L_1, L_2, L_3 – номера инцидентных отрезков в принимающем процессе с соответствующими интенсивностями IL_1, IL_2, IL_3 , а P_1, P_2, P_3 – номера инцидентных узлов в принимающем процессе с соответствующими интенсивностями IP_1, IP_2, IP_3 . Передача сообщения в нужную нить обеспечивается выбором тега.

Для описания внешней границы $(OBS_0^i, OBS_1^i, OBS_2^i)$ можно пользоваться той же структурой данных, что применялась для описания границы (BS_0, BS_1, BS_2) в последовательном алгоритме. Однако, в отличие от последовательного алгоритма, внешняя граница $(OBS_0^i, OBS_1^i, OBS_2^i)$ не является замкнутой. Край двумерного комплекса $(OBS_0^i, OBS_1^i, OBS_2^i)$ является одномерным комплексом $(EOBS_0^i, EOBS_1^i)$ ($EOBS$ – extreme outer boundary simplex), каждый отрезок которого входит ровно в два процесса. Поэтому для склейки краев внешних границ вводится дополнительная структура данных, задаваемая списком крайних внешних граничных отрезков.

- Объект «**крайний внешний граничный отрезок**» имеет одно поле под указатель на соответствующий внешний граничный отрезок из OBS_1^i . Два поля под указатели на инцидентные внешние граничные узлы из OBS_0^i . Четыре поля под целые числа: номер j смежного процесса, номер рассматриваемого внешнего граничного отрезка в списке OBS_1^j смежного процесса, а также номера инцидентных внешних граничных узлов из OBS_0^j , записанные в правильном порядке.

Алгоритм 2

Вход: В i -ом вычислительном узле задана неструктурированная сетка $(S_0^i, S_1^i, S_2^i, S_3^i)$ с границами $(CBS_0^i, CBS_1^i, CBS_2^i)$ и $(OBS_0^i, OBS_1^i, OBS_2^i)$, $i = 1, \dots, P$. Задана кубатурная формула (7) с узлами $\Omega_1, \dots, \Omega_N$. На внешней границе $(OBS_0^i, OBS_1^i, OBS_2^i)$ поставлены граничные условия (4): для всех угловых направлений заданы интенсивности в сеточных элементах внешней освещенной границы.

Выход: Получено решение задачи переноса (1), (4) интерполяционно-характеристическим методом в соответствии с правильным порядком обхода ячеек: интегральные величины (5) рассчитаны во всех элементах сеток $(S_0^i, S_1^i, S_2^i, S_3^i)$, интенсивности излучения вычислены во всех элементах внешних границ $(OBS_0^i, OBS_1^i, OBS_2^i)$ и во всех направлениях $\Omega_1, \dots, \Omega_N$. Либо получен сигнал о наличии цикла в графе G_{Ω_s} при некотором значении s .

(Этот псевдокод исполняется на i -ом MPI-узле, $i = 1, \dots, P$)

Для всех s от 1 до N :

(Цикл угловых направлений распараллеливается по ядрам MPI-узла)

Инициализировать массивы $I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$ для сеточных интенсивностей;

Инициализировать битовый массив $\text{ord}^{i,s}$ для упорядочения;

Для всех треугольников $t(c \rightarrow c') \in S_2^i$:

Присвоить $\text{ord}^{i,s}[t] := 1$;

Вычислить вектор нормали n к треугольнику t , направленный из ячейки c или в ячейку c' ;

Если $\langle n, \Omega_i \rangle < 0$:

Присвоить $\text{ord}^{i,s}[t] := 0$;

Инициализировать массив $\text{deg}^{i,s}$ входящих степеней ячеек;

Для всех ячеек $c \in S_3^i$:

Присвоить $\text{deg}^{i,s}[c] := 0$;

Для всех треугольников $t(c' \rightarrow c'')$, инцидентных ячейке c :

Если $(c'' = c$ и $\text{ord}^{i,s}[t] = 1)$ или $(c' = c$ и $\text{ord}^{i,s}[t] = 0)$:

Увеличить $\text{deg}^{i,s}[c]$ на единицу;

Для всех внешних граничных треугольников $obt \in OBS_2^i$:

Рассмотреть соответствующий obt треугольник $t(c \rightarrow c') \in S_2^i$;

Если $(c = \text{NULL}$ и $\text{ord}^{i,s}[t] = 1)$ или $(c' = \text{NULL}$ и $\text{ord}^{i,s}[t] = 0)$:

Записать только оригинальные данные в массивы

$I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$ (повторно полученные данные не перезаписывать)
из элементов граничного треугольника obt для направления Ω_s ;

Учесть вклад от оригинальных данных в расчет интегральных
величин (5) в элементах треугольника t ;

Если $c = \text{NULL}$:

Уменьшить $\text{deg}[c']$ на единицу;

Иначе:

Уменьшить $\text{deg}[c]$ на единицу;

Инициализировать очередь $Q^{i,s}$ для указателей на разрешимые ячейки;

Для всех ячеек $c \in S_3^i$:

Если $\text{deg}^{i,s}[c] = 0$:

Поместить указатель на c в конец очереди $Q^{i,s}$;

Инициализировать счетчик $K^{i,s} := 0$ для разрешенных ячеек;

Пока не будет достигнута **точка выхода** из цикла, повторять:

Если очередь $Q^{i,s}$ не пуста:

Передать 1 корневому процессу в сообщении с тэгом s ;

Достать ячейку c из начала очереди $Q^{i,s}$;

Разрешить ячейку c ;

Записать только оригинальные данные в массивы

$I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$ (повторно полученные данные не перезаписывать);

Учесть вклад от оригинальных данных
в расчет интегральных величин (5) в элементах
неосвещенных треугольников ячейки c ;

Увеличить $K^{i,s}$ на единицу;

Для всех треугольников $t(c' \rightarrow c'')$, инцидентных ячейке c :

Если $c'' \neq \text{NULL}$ и $\text{ord}^{i,s}[t] = 1$:

Уменьшить $\text{deg}^{i,s}[c'']$ на единицу;

Если $\text{deg}^{i,s}[c''] = 0$:

Поместить указатель на c'' в конец очереди $Q^{i,s}$;

Иначе если $c' \neq \text{NULL}$ и $\text{ord}^{i,s}[t] = 0$:

Уменьшить $\text{deg}^{i,s}[c']$ на единицу;

Если $\text{deg}^{i,s}[c'] = 0$:

Поместить указатель на c' в конец очереди $Q^{i,s}$;

Иначе если ($c'' = \text{NULL}$ и $\text{ord}^{i,s}[t] = 1$)

или ($c' = \text{NULL}$ и $\text{ord}^{i,s}[t] = 0$):

Произвести поиск смежного граничного треугольника $cbt \in CBS_2^i$, которому соответствует треугольник t ;

Если cbt найден:

Отправить в смежный процесс сообщение с тэгом s и данными, соответствующими элементам треугольника t из массивов $I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$;

Иначе:

Произвести поиск внешнего граничного треугольника $obt \in OBS_2^i$, которому соответствует треугольник t ;

Записать данные в элементы внешнего граничного треугольника obt для направления Ω_s из массивов $I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$;

Если очередь $Q^{i,s}$ пуста:

Передать 0 координирующему процессу в сообщении с тэгом s ;

Принять сообщение с тэгом s от любого источника;

Если пришло сообщение от смежного процесса:

Записать только оригинальные данные из сообщения в массивы $I_0^{i,s}, I_1^{i,s}, I_2^{i,s}$

(повторно полученные данные не перезаписывать);

Учесть вклад от оригинальных данных в расчет интегральных величин (5), переданных в треугольник $t(c \rightarrow c')$;

Если $c = \text{NULL}$:

Уменьшить $\text{deg}^{i,s}[c']$ на единицу;

Если $\text{deg}^{i,s}[c'] = 0$:

Поместить указатель на c' в конец очереди $Q^{i,s}$;

Иначе если $c' = \text{NULL}$:

Уменьшить $\text{deg}^{i,s}[c]$ на единицу;

Если $\text{deg}^{i,s}[c] = 0$:

Поместить указатель на c в конец очереди $Q^{i,s}$;

Если пришло сообщение от корневого процесса:

Если $K^{i,s} \neq |S_3^i|$:

Передать -1 координирующему процессу

В сообщении с тэгом s ;

Точка выхода из цикла;

Конец алгоритма;

(Этот псевдокод исполняется на корневом процессе: нулевом MPI-узле)

Инициализировать двумерную битовую таблицу T ,
заполненную единицами, имеющую P строк (P – количество подобластей) и
 N столбцов (N – количество угловых направлений);

Инициализировать количество обработавших направлений $n := 0$;

Пока $n < N$:

 Принять сообщение из любого источника;

 Если получено -1:

 Выяснить тэг сообщения s ;

 Сообщить, что имеется цикл в графе G_{Ω_s} ;

 Иначе:

 Записать в $T[i][s]$ полученное сообщение

 от i -го процесса с тэгом s ;

 Если дизъюнкция элементов s -го столбца таблицы T равна 0:

 Передать всем рабочим процессам сообщение с тэгом s ;

 Увеличить n на единицу;

Конец алгоритма;

Применимость интерполяционно-характеристического метода

Вопреки самым лучшим ожиданиям, существуют неструктурированные сетки из тетраэдров, на которых невозможно произвести решение задачи переноса (1), (4), применяя метод характеристик. Как отмечалось выше, эта невозможность связана с наличием цикла в ориентированном сеточном графе G_{Ω} для некоторого направления излучения Ω . Ниже приводится пример конфигурации ячеек с циклом.

Рассмотрим правильную семиугольную призму $A_0, \dots, A_6, B_0, \dots, B_6$, вершины которой задаются в декартовых координатах следующим образом:

$$A_k = \left(\cos \frac{2\pi k}{7}, \sin \frac{2\pi k}{7}, 0 \right), B_k = \left(\cos \frac{2\pi k}{7}, \sin \frac{2\pi k}{7}, 1 \right).$$

Пусть t_i – тетраэдр с узлами $A_i, A_{i+1}, B_{i+1}, B_{i+2}$, а u_i – тетраэдр с узлами $A_i, A_{i+1}, B_{i+2}, B_{i+3}$, где суммирование в индексах производится в циклической группе \mathbb{Z}_7 . На рис. 2 изображены тетраэдры t_i и u_i в семиугольной призме.

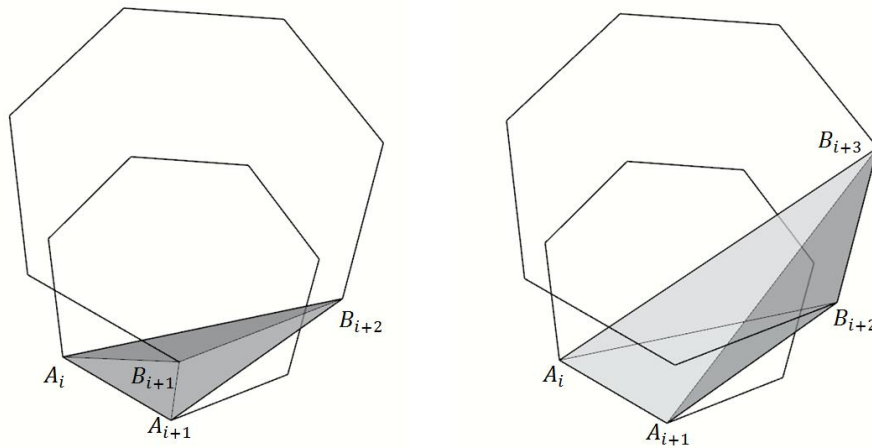


Рис. 2. Тетраэдры t_i и u_i

Тетраэдры t_i и u_i имеют общую треугольную грань с узлами A_i, A_{i+1}, B_{i+2} , а u_i и t_{i+1} — с узлами $A_{i+1}, B_{i+2}, B_{i+3}$. Набор ячеек $t_0, u_0, \dots, t_6, u_6$ вместе с гранями меньших размерностей удовлетворяет всем требованиям к неструктурированной сетке. На рис. 3 изображается весь комплекс из ячеек $t_0, u_0, \dots, t_6, u_6$. Однако для направления излучения $\Omega = (0, 0, -1)$ последовательность ячеек $t_0, u_0, \dots, t_6, u_6, t_0$ образует цикл в графе G_Ω .

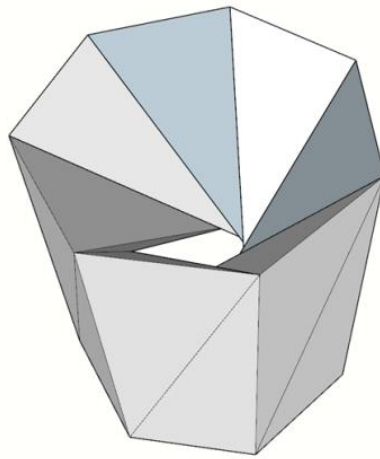


Рис. 3. Пример неструктурированной сетки с циклом

Тем не менее, широкий класс неструктурированных сеток позволяет применять интерполяционно-характеристический метод. К этому классу относятся, например, сетки Делоне. Напомним, что сетка удовлетворяет критерию Делоне, если и только если внутренность описанного шара каждого тетраэдра не содержит узлов сетки.

В работе [12] показано, что для графа G_Ω в случае сетки Делоне выполнено:

стрелка $e \in E_\Omega$ направлена из вершины $v_1 \in V$ в вершину $v_2 \in V$ тогда и только тогда, когда $\langle r_{12}, \Omega \rangle > 0$, где r_{12} – вектор, соединяющий центр описанной сферы ячейки v_1 с центром описанной сферы ячейки v_2 . Этот факт сразу же влечет ацикличность графа G_Ω , а также то, что обход ячеек осуществляется в соответствии с порядком проекций центров их описанных сфер на направление Ω .

В работе [13] доказывается ацикличность регулярных сеток. Регулярными называются сетки, получающиеся проектированием границы четырехмерного выпуклого многогранника на трехмерное гиперпространство. Сетки Делоне являются частным случаем регулярных сеток. В книге [14] подробно изучается класс регулярных сеток.

Заключение

В предлагаемой работе описана реализация алгоритма обхода ячеек неструктурированной сетки при решении уравнения переноса излучения интерполяционно-характеристическим методом. Этот алгоритм основан на топологической сортировке ориентированного графа, поэтому является наиболее эффективным из всех возможных. Приведена параллельная реализация алгоритма обхода в случае, когда расчетная сетка разбита на отдельные области, хранимые на разных узлах вычислительного кластера.

Библиографический список

[1] *Четверушкин Б.Н.* Математическое моделирование задач динамики излучающего газа – М.: Наука. Главная редакция физико-математической литературы, 1985. – 304 с.

[2] *Аристова Е.Н., Астафуров Г.О.* Характеристическая схема для решения уравнения переноса на неструктурированной сетке с барицентрической интерполяцией // Математическое моделирование, 30:9, 2018. – С. 33-50.

E.N. Aristova, G.O. Astafurov Harakteristicheskaya skhema dlya resheniya uravneniya perenosa na nestrukturirovannoj setke s baricentricheskoj interpolyaciej // Matematicheskoe modelirovanie, 30:9, 2018. – pp. 33-50.

[3] *Николаева О.В.* Нодальная сеточная схема для уравнения переноса на неструктурированной тетраэдральной сетке // Математическое моделирование, 27:5, 2015. – С. 80–96.

http://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=mm&paperid=3601&option_lang=rus

O.V. Nikolaeva Nodal scheme for the radiation transport equation on an unstructured tetrahedral mesh // Mathematical models and computer simulations, 7:6, 2015. – pp. 581-592.

<https://link.springer.com/article/10.1134%2FS2070048215060071>

[4] *Аристова Е.Н., Астафуров Г.О.* Метод коротких характеристик второго порядка для решения уравнения переноса на сетке из тетраэдров // Математическое моделирование, 28:7, 2016. – С. 20–30.

<http://www.mathnet.ru/links/a4c8c106df1a594e005f4c406d48b2b7/mm3745.pdf>

E.N. Aristova, G.O. Astafurov Second-order short characteristic method for solving the transport equation on a tetrahedron mesh // Mathematical models and computer simulations, 9:1, 2017. – pp. 40-47.

<https://link.springer.com/article/10.1134%2F52070048217010045>

[5] *Мысовских И.П.* Интерполяционные кубатурные формулы – М.: Наука. Главная редакция физико-математической литературы, 1981. – 336 с.

[6] *Соболев С.Л., Васкевич В.Л.* Кубатурные формулы – Новосибирск: Издательство Института математики, 1996. – 484 с.

[7] *Яковлев Е.И.* Вычислительная топология – Нижний Новгород: Издательство Нижегородского университета, 2003. – 198 с.

[8] *Яковлев Е.И., Изумнов Л.А.* Методы вычисления на супер-ЭВМ топологических характеристик триангулированных объектов – Нижний Новгород: Издательство Нижегородского университета, 2014. – 85 с.

[9] *Дасгупта С., Пападимитриу Х., Вазирани У.* Алгоритмы – М.: МЦНМО, 2014. – 320 с.

S. Dasgupta, C. Papadimitriou, U. Vazirani Algorithms. – McGraw-Hill, 2006. – 336 p.

[10] *Антонов А.С.* Параллельно программирование с использованием технологии MPI – М.: Издательство Московского университета, 2004. – 72 с.

[11] *Антонов А.С.* Параллельное программирование с использованием технологии OpenMP – М.: Издательство Московского университета, 2009. – 76 с.

[12] *Скалько Ю.И., Карасев Р.Н., Акопян А.В., Цыбулин И.В., Мендель М.А.* Маршевый алгоритм решения задачи переноса излучения методом коротких характеристик // Компьютерные исследования и моделирование, 6:2, 2014. – С. 203-215.

<http://crm.ics.org.ru/journal/article/2138/>

Y.I. Skalko, R.N. Karasev, A.V. Akopyan, I.V. Tsybulin, M.A. Mendel Marshevyj algoritm resheniya zadachi perenosa izlucheniya metodom korotkih harakteristik // Komp'yuternye issledovaniya i modelirovanie, 6:2, 2014. – pp 203-215.

[13] *Edelsbrunner H.* An acyclicity theorem for cell complexes in d dimension // Combinatorica, 10:3, 1990. – pp. 251-260.

<https://link.springer.com/article/10.1007/BF02122779>

[14] *De Lorea J., Rambau J., Santos F.* Triangulations. Structures for algorithms and applications – Springer, 2010. – 535 p.