



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

Препринты ИПМ • Препринт № 245 за 2018 г.



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

**Краснов М.М., Ладонкина М.Е.,
Тишкин В.Ф.**

Реализация разрывного
метода Галёркина в
программном комплексе
DGM

Рекомендуемая форма библиографической ссылки: Краснов М.М., Ладонкина М.Е., Тишкин В.Ф. Реализация разрывного метода Галёркина в программном комплексе DGM // Препринты ИПМ им. М.В.Келдыша. 2018. № 245. 31 с. doi:[10.20948/prepr-2018-245](https://doi.org/10.20948/prepr-2018-245)
URL: <http://library.keldysh.ru/preprint.asp?id=2018-245>

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

М. М. Краснов, М. Е. Ладонкина, В. Ф. Тишкин

**Реализация разрывного метода Галёркина
в программном комплексе DGM**

Москва— 2018

Краснов М. М., Ладонкина М. Е., Тишкин В. Ф.

Реализация разрывного метода Галёркина в программном комплексе DGM

Излагаются математические основы разрывного метода Галёркина, на которых базируется реализация программного комплекса DGM. В данном программном комплексе реализован модальный подход к методу Галёркина с разрывными базисными функциями для решения уравнений Навье-Стокса на сетках произвольной структуры. Особое внимание уделено реализации лимитера Кокбурна для ячеек произвольной формы.

Ключевые слова: разрывный метод Галёркина, уравнение Навье-Стокса, лимитеры, программный комплекс DGM

Mikhail Mikhailovich Krasnov, Marina Evgenyevna Ladonkina, Vladimir Fedorovich Tishkin

Implementation of the Galerkin discontinuous method in the DGM software package

The mathematical foundations of the discontinuous Galerkin method are presented, on which the implementation of the DGM software package is based. This software package implements a modal approach to the Galerkin method with discontinuous basis functions for solving the Navier-Stokes equations on grids of arbitrary structure. Particular attention is paid to the implementation of the Cockburn limiter for cells of arbitrary shape.

Key words: Discontinuous Galerkin method, Navier-Stokes equation, limiters, DGM software package

Работа выполнена в рамках госзадания ИПМ им. М.В. Келдыша.

Оглавление

Введение	3
Постановка задачи	6
Разрывный метод Галёркина	8
Граничные условия	17
Лимитеры	19
Библиографический список	27

Введение

В настоящее время для решения широкого класса прикладных задач математической физики необходимо использовать численные методы высокой точности. Примером метода, обеспечивающего высокую точность, является метод Галёркина с разрывными базисными функциями (РМГ) или Discontinuous Galerkin Method (DGM). Данный метод обладает рядом достоинств, присущих как конечно-элементным, так и конечно-разностным аппроксимациям. В частности, он обеспечивает заданный порядок точности, причем на неструктурированных сетках [1], может использоваться для сеток с произвольной формой ячеек и теоретически обоснован. Это особенно актуально для решения сложных разномасштабных задач, в которых получить решение только измельчением сетки и методами первого порядка точности недостаточно.

Метод Галёркина с разрывными базисными функциями был предложен более сорока лет назад для решения уравнения переноса нейтронов [2]. Но особенно активно РМГ начал развиваться в последние два десятилетия. Каждый год появляется огромное количество работ, посвященных исследованию и применению разрывного метода Галёркина. На последней конференции в рамках ECCOMAS: European Conference on High Order Nonlinear Numerical Methods for Evolutionary PDEs: Theory and Applications HONOM 2017 [3], проходившей в марте 2017 г. в Штутгарте, более половины докладов были посвящены применению разрывного метода Галёркина.

Существует два подхода для повышения точности получаемого решения. Один из них – это измельчение сетки в областях существующих особенностей решения, второй подход – повышение порядка точности схемы. Применение разрывного метода Галёркина позволяет использовать сразу оба подхода – как повышение порядка точности метода за счет повышения порядка используемых полиномов, так и локальное измельчение сетки (так называемая hp-адаптация) [4, 5, 6, 7].

В рамках реализации разрывного метода Галёркина приближенное решение определяется в виде проекции вектора консервативных переменных на пространство полиномов степени p с зависящими от времени коэффициентами.

Как принято при решении уравнений второго порядка разрывным методом Галёркина, уравнения Навье-Стокса записываются в виде системы уравнений первого порядка, и решение происходит в два этапа. На первом этапе вычисляются компоненты градиента температуры и тензора вязких напряжений, аппроксимация которых, как и аппроксимация решения, в пределах ячейки сетки находится в виде полиномов степени s с зависящими от времени коэффициентами. На границе элемента потоковые значения величин определяются по некоторому правилу от значений внутри элемента и от значений в соседней к данному элементу ячейке.

На следующем этапе определяются компоненты вектора консервативных переменных. При этом конвективные потоки могут быть рассчитаны при помощи

различных вариантов точного или приближенного решения задачи Римана, диффузионные потоки на границе элемента также могут быть рассчитаны различными способами, подробный анализ которых проведен в работе [8]. Для вычисления интегралов используются квадратурные формулы нужного порядка [9]. Данный двухэтапный подход позволяет вычислять градиенты с тем же порядком точности, что и консервативные переменные. При этом сохраняется компактность шаблона схемы.

Одним из важных вопросов при реализации метода является выбор сетки, на которой ищется решение. Несомненным преимуществом РМГ является возможность его применения на сетках произвольной структуры. В настоящее время разрывный метод Галёркина хорошо разработан для структурированных сеток, см., например, [10], но данный метод менее разработан для неструктурированных сеток. Однако существуют удачные программные реализации РМГ для решения трехмерных задач на неструктурированных сетках, содержащих элементы только одного типа (тетраэдральные [10, 11, 12, 13] или гексаэдральные [5]), а также для сеток произвольной структуры [14]. Очевидным недостатком метода является его чрезвычайно высокая вычислительная стоимость, которая покрывается компактным шаблоном и, следовательно, созданием эффективных параллельных программных комплексов.

Помимо множества преимуществ применения разрывного метода Галёркина, он имеет некоторые сложности реализации. Во-первых, для обеспечения монотонности решения, полученного данным методом, необходимо вводить так называемые ограничители наклона, или лимитеры, в особенности в том случае, если решение содержит сильные разрывы. Наиболее широко используемым является классический лимитер Кокбурна [15]. Идея данного лимитера легко реализуется в многомерном случае на сетках произвольной структуры. Однако, данный лимитер, как и все TVD лимитеры, снижает точность получаемого решения. В последнее время активно развиваются различные подходы к решению этой проблемы. Один из подходов к созданию лимитера повышенного порядка точности предложен в работах Кривоносовой [16, 17]. Но данный лимитер хорошо работает только на структурированных сетках. Другой подход к созданию лимитера повышенного порядка точности заключается в использовании WENO-ограничителя [18]-[24]. В работах [17], [25] предложены лимитеры, не прибегающие к использованию $\min\text{mod}$ процедуры, что является большим преимуществом данных лимитеров. В работе [25] предложен сглаживающий оператор нового типа на основе усреднения решений, учитывающий скорость изменения решения и скорость изменения его производных. В работе [17] дополнительно предложен анализатор разрывов решения. Ещё один принципиально другой способ монотонизации решения основан на введении искусственной вязкости в численную схему [26, 27], этот подход предполагает использование эмпирических констант, что усложняет его применение к решению реальных задач. В работе [28] предложена модификация

РМГ для двумерного случая, обеспечивающая возможность плавного перехода от схемы высокого порядка точности к монотонной схеме первого порядка точности в областях сингулярности течения. Наиболее простая реализация лимитера предложена в работе [29].

Также важным вопросом при реализации РМГ является вопрос о выборе дискретной аппроксимации потоков для вязких членов уравнения Навье-Стокса. В настоящее время существует несколько видов таких аппроксимаций, наиболее часто используемые в реальных расчетах [30, 31, 32] были исследованы в работе [33]. Однако вопрос оптимального выбора таких аппроксимаций для рассматриваемых задач остается открытым.

При повышении порядка точности схемы возникает жесткое ограничение на шаг по времени. Первоначально при расчетах разрывным методом Галёркина интегрирование по времени проводилось явными многошаговыми схемами Рунге-Кутты высокого порядка [15], [34], [35]. Но наиболее эффективный способ проведения расчетов состоит в использовании неявных методов интегрирования по времени, целью которого является ослабление ограничения на шаг по времени [36] - [44]. В настоящий момент известны программные реализации РМГ неявным методом для моделирования несжимаемых течений [45] и для решения уравнений Навье-Стокса [14].

Таким образом, для успешной реализации РМГ на трехмерных неструктурированных сетках необходимо сосредоточить внимание на несколько моментах. Во-первых, на построении лимитирующих функций, во-вторых, на выборе наилучших дискретных аппроксимаций диффузионных потоков, и, наконец, на применении неявных и итерационных методов решения полученных дифференциально-разностных уравнений.

Как известно, разрывный метод Галёркина обладает существенной вычислительной сложностью, поэтому встает вопрос о максимально эффективном использовании всех возможностей вычислительной техники. В настоящее время в мировых исследовательских центрах, занимающихся этой проблемой, ведутся работы по распараллеливанию реализаций РМГ на суперЭВМ [46, 47, 48]. Динамика развития высокопроизводительной вычислительной техники диктует создание программных комплексов, сравнительно легко адаптируемых для работы на различных (в том числе гибридных) параллельных архитектурах. При создании программ, основанных на численных методах высокой точности, таких как РМГ, необходимо использовать новые подходы к программированию. В рамках данной работы при решении уравнений Навье-Стокса разрывным методом Галёркина использован новый сеточно-операторный подход к программированию задач математической физики, позволяющий компактно записывать и эффективно реализовывать математические формулы, единообразно реализовывать подход на разных типах сеток и для различных вычислительных архитектур, в том числе для графических ускорителей CUDA [13, 49, 50].

В данной работе приводится математическое обоснование разрывного метода

Галёркина и даётся подробное описание его реализации в программном комплексе DGM [12, 13].

Постановка задачи

Введём обозначения, которые будут в дальнейшем использованы. Мы будем оперировать с величинами четырёх типов: числовыми (например, плотность, температура, давление, энергия) и векторными трёх видов. Первый – это классические вектора (например, скорость, импульс), число компонент которых равно размерности пространства, для которого ведутся расчёты (2 или 3). Такие вектора будем обозначать, как и принято, со стрелочкой сверху (например, \vec{v}). Второй вид векторов – это т.н. вектора консервативных переменных, используемых при решении уравнения Навье-Стокса. Компонентами данного вектора являются плотность, компоненты импульса и энергия. В зависимости от размерности пространства таких переменных получается или 4, или 5. Для того чтобы отличать такие вектора от обычных векторов, будем их обозначать с чёрточкой сверху (например, \bar{U}). Кроме того, мы будем оперировать обычными векторами, компонентами которых будут не числа, а вектора консервативных переменных. Это т.н. вектора потоков. Такие вектора мы по-прежнему будем обозначать со стрелочкой сверху, но сами переменные будем выделять полужирным шрифтом (например, \vec{F}).

Итак, мы определили величины четырёх типов: числа, вектора чисел, вектора консервативных переменных и вектора потоков.

Обозначим скалярное произведение двух векторов \vec{f} и \vec{g} через (\vec{f}, \vec{g}) .

Математические операторы, с которыми мы будем работать, могут быть скалярными или векторными. Скалярные математические операторы – это операторы частных производных по времени ∂_t и по координатам ∂_x , ∂_y и ∂_z . Векторный оператор один – это оператор $\vec{\nabla}$ (набла): $\vec{\nabla} = (\partial_x, \partial_y, \partial_z)^T$.

Пусть переменная $\bar{U}(\vec{r}, t)$ обозначает вектор консервативных переменных, а $\vec{F}(\bar{U}, \vec{\nabla}\bar{U})$ – векторную потоковую функцию этих консервативных переменных. Рассмотрим трёхмерное уравнение Навье-Стокса, записанное в консервативной форме

$$\partial_t \bar{U} + (\vec{\nabla}, \vec{F}) = 0 \quad (1)$$

и дополненное подходящими начально-краевыми условиями.

Потоковая функция \vec{F} в общем случае состоит из трёх аддитивных компонент: невязкой (Эйлеровой), вязкой и теплопроводной:

$$\vec{F} = \vec{F}^E - \vec{F}^V - \vec{F}^T, \quad (2)$$

при этом вязкая и теплопроводная компоненты (одна или обе) могут отсутствовать.

Консервативные переменные \bar{U} заданы в виде:

$$\bar{U} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ E \end{pmatrix},$$

а Эйлера потоковая функция $\vec{F}^E = (\bar{F}_x^E, \bar{F}_y^E, \bar{F}_z^E)^T$ задаётся формулами:

$$\bar{F}_x^E = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ \rho v_x v_z \\ (E + p)v_x \end{pmatrix}, \bar{F}_y^E = \begin{pmatrix} \rho v_y \\ \rho v_y v_x \\ \rho v_y^2 + p \\ \rho v_y v_z \\ (E + p)v_y \end{pmatrix}, \bar{F}_z^E = \begin{pmatrix} \rho v_z \\ \rho v_z v_x \\ \rho v_z v_y \\ \rho v_z^2 + p \\ (E + p)v_z \end{pmatrix},$$

где ρ – плотность жидкости, $\vec{v} = (v_x, v_y, v_z)^T$ – вектор скорости, p – давление и $E = \rho(e + v^2/2)$ – полная энергия.

Полагая, что жидкость подчиняется произвольным уравнениям состояния, p можно записать как $p = p(\rho, e)$.

Функция вязкого потока $\vec{F}^V = (\bar{F}_x^V, \bar{F}_y^V, \bar{F}_z^V)^T$ задаётся формулами:

$$\bar{F}_x^V = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ (\vec{v}, \vec{\tau}_x) \end{pmatrix}, \bar{F}_y^V = \begin{pmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ (\vec{v}, \vec{\tau}_y) \end{pmatrix}, \bar{F}_z^V = \begin{pmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ (\vec{v}, \vec{\tau}_z) \end{pmatrix},$$

где $\hat{\tau}$ – тензор вязких напряжений:

$$\hat{\tau} = \|\vec{\tau}_x \vec{\tau}_y \vec{\tau}_z\| = \begin{vmatrix} \tau_{xx} \tau_{yx} \tau_{zx} \\ \tau_{xy} \tau_{yy} \tau_{zy} \\ \tau_{xz} \tau_{yz} \tau_{zz} \end{vmatrix},$$

который вычисляется по формуле:

$$\hat{\tau} = \mu \left[\vec{\nabla} \otimes \vec{v} + \left(\vec{\nabla} \otimes \vec{v} \right)^T \right] + \left(\lambda - \frac{2}{3}\mu \right) \hat{\delta} \cdot \left(\vec{\nabla}, \vec{v} \right), \quad (3)$$

где $i, j = x, y, z$; μ – коэффициент динамической вязкости, λ – объёмная вязкость, а $\hat{\delta}$ – символ Кронекера (единичный тензор):

$$\delta_{ij} = \begin{cases} 1, & \text{если } i = j; \\ 0, & \text{если } i \neq j. \end{cases}$$

Знаком \otimes обозначено тензорное произведение векторов:

$$\vec{f} \otimes \vec{g} = \vec{f} \cdot \vec{g}^T.$$

В правой части последнего уравнения вектора \vec{f} и \vec{g} рассматриваются как матрицы с одним столбцом, а знак « \cdot » обозначает обычное умножение матриц.

Вектор потоковой функции теплопроводности $\vec{F}^T = (\bar{F}_x^T, \bar{F}_y^T, \bar{F}_z^T)^T$ запишем как

$$\bar{F}_x^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ q_x \end{pmatrix}, \bar{F}_y^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ q_y \end{pmatrix}, \bar{F}_z^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ q_z \end{pmatrix},$$

где

$$\vec{q} = \hat{\kappa} \cdot \vec{\nabla} T = (q_x, q_y, q_z)^T, \quad (4)$$

$\hat{\kappa}$ – тензор удельной теплопроводности, а $T = T(\vec{r})$ – температура.

Разрывный метод Галёркина

Перейдём к описанию разрывного метода Галёркина. Покроем область Ω , на которой ищется решение, сеткой $K_h : \Omega = \cup K$. Определим линейное пространство E_K с базисом:

$$\Phi_{K,i}(\vec{r}) = \begin{cases} \varphi_{K,i}, & \text{если } \vec{r} \in K; \\ 0, & \text{если } \vec{r} \notin K. \end{cases} \quad (5)$$

$i = \overline{0, N-1}$, где N – размерность пространства базисных функций. В качестве базисных функций для данной ячейки K выберем всевозможные многочлены вида:

$$\varphi_{K,i}(\vec{r}) = \left(\frac{x - x_c}{\Delta x} \right)^{\alpha_{K,i}} \cdot \left(\frac{y - y_c}{\Delta y} \right)^{\beta_{K,i}} \cdot \left(\frac{z - z_c}{\Delta z} \right)^{\gamma_{K,i}},$$

такие, что сумма степеней $\alpha_{K,i} + \beta_{K,i} + \gamma_{K,i}$ не превышает некоторого заданного числа p . В реальных расчётах максимальная степень многочленов p обычно выбирается небольшой (от 0 до 2). Здесь x_c, y_c, z_c – координаты центра масс ячейки, а $\Delta x, \Delta y, \Delta z$ – характерные размеры ячейки. Для треугольников и тетраэдров центр масс совпадает с геометрическим центром (средним арифметическим координат вершин), для ячеек других типов эти две точки могут, вообще говоря, не совпадать.

В линейном пространстве базисных функций в каждой ячейке у нас появляются свои вектора. Такие вектора будем обозначать с волнистой чертой сверху (например, \tilde{q}).

Будем искать решение в линейном пространстве E_K :

$$\bar{U}(\vec{r}, t) = \sum_K \sum_{i=0}^{N-1} \bar{U}_{K,i}(t) \Phi_{K,i}(\vec{r}), \quad (6)$$

где $\bar{U}_{K,i}(t)$ – коэффициенты разложения численного решения, а $\Phi_{K,i}(\vec{r})$ – соответствующая базисная функция.

Умножая равнение (1) на произвольную пробную функцию $\omega \in E_K$

$$\omega(\vec{r}, t) = \sum_K \sum_{i=0}^{N-1} \omega_{K,i}(t) \Phi_{K,i}(\vec{r}), \quad (7)$$

где $\omega_{K,i}(t)$ – некоторые коэффициенты, и проинтегрировав по области Ω , мы получим формулировку разрывного метода Галёркина:

$$\int_{\Omega} \omega \partial_t \bar{U} d\Omega + \int_{\Omega} \omega \left(\vec{\nabla}, \vec{\mathbf{F}} \right) d\Omega = 0, \forall \omega. \quad (8)$$

Если переписать формулу (8) с использованием (7) в виде

$$\sum_K \sum_{i=0}^{N-1} \omega_{K,i} \left[\int_{\Omega} \Phi_{K,i} \partial_t \bar{U} d\Omega + \int_{\Omega} \Phi_{K,i} \left(\vec{\nabla}, \vec{\mathbf{F}} \right) d\Omega \right] = 0, \forall \omega, \quad (9)$$

то становится ясно, что для выполнения (9) достаточно потребовать, чтобы

$$\int_{\Omega} \Phi_{K,i} \partial_t \bar{U} d\Omega + \int_{\Omega} \Phi_{K,i} \left(\vec{\nabla}, \vec{\mathbf{F}} \right) d\Omega = 0, \forall K, i. \quad (10)$$

В работе [51] показано, что если использовать сглаживание разрывных функций, входящих в интегралы

$$\int_K \Phi_{K,i} \partial_t \bar{U} d\Omega + \int_K \Phi_{K,i} \left(\vec{\nabla}, \vec{\mathbf{F}} \right) d\Omega = 0, \forall K, i. \quad (11)$$

то в пределе, при уменьшении интервала сглаживания до нуля, интегралы существуют при условии, что на границе ∂K элемента K сглаженная функция $\vec{\mathbf{F}}$ имеет определенное значение $\vec{\mathbf{F}}^*$. В этом случае, применяя формулу интегрирования по частям ко второму слагаемому уравнения (8), содержащему дивергенцию функции потока, получим:

$$\int_{\Omega} \omega \partial_t \bar{U} d\Omega + \int_{\Omega} (\vec{\nabla}, \omega \vec{\mathbf{F}}) d\Omega - \int_{\Omega} (\vec{\nabla} \omega, \vec{\mathbf{F}}) d\Omega = 0, \forall \omega.$$

Так как последнее уравнение справедливо для произвольной функции $\omega \in E_K$, то оно справедливо для каждой базисной функции $\Phi_{K,j}(\vec{r})$. Таким образом, для каждой функции $\Phi_{K,j}$ мы получим:

$$\int_{\Omega} \Phi_{K,j} \partial_t \bar{U} d\Omega + \int_{\Omega} (\vec{\nabla}, \Phi_{K,j} \vec{\mathbf{F}}) d\Omega - \int_{\Omega} (\vec{\nabla} \Phi_{K,j}, \vec{\mathbf{F}}) d\Omega = 0.$$

Учитывая представление базисных функций $\Phi_{K,j}(\vec{r})$ в виде (5), для каждой ячейки K получим:

$$\int_K \varphi_{K,j} \partial_t \bar{U} d\Omega + \int_K (\vec{\nabla}, \varphi_{K,j} \vec{\mathbf{F}}) d\Omega - \int_K (\vec{\nabla} \varphi_{K,j}, \vec{\mathbf{F}}) d\Omega = 0. \quad (12)$$

Воспользуемся формулой Остроградского-Гаусса:

$$\int_V (\vec{\nabla}, \vec{f}) d\Omega = \oint_S (\vec{f}, \vec{n})^\Gamma d\sigma. \quad (13)$$

Знак Γ после скалярного произведения в правой части этого уравнения (потока через границу) означает, что если векторная функция \vec{f} на границе претерпевает разрыв, то значение этого потока нужно вычислять специальным образом с учётом значения функции с другой стороны границы (в соседней ячейке). Применив эту формулу ко второму слагаемому формулы (12), мы получим ослабленную формулировку разрывного метода Галёркина:

$$\int_K \varphi_j \partial_t \bar{U} d\Omega + \oint_{\partial K} \varphi_j (\vec{\mathbf{F}}, \vec{n})^\Gamma d\sigma - \int_K (\vec{\nabla} \varphi_j, \vec{\mathbf{F}}) d\Omega = 0, j = \overline{0, N-1}. \quad (14)$$

Здесь и в дальнейшем для сокращения записи будем опускать индекс K в обозначении $\varphi_{K,j}(\vec{r})$. При вычислении поверхностного интеграла в последней формуле базисная функция φ_j берётся из текущей ячейки (по границе которой берётся интеграл).

В каждой ячейке разложение (6) принимает вид:

$$\bar{U}(\vec{r}, t) = \sum_{i=0}^{N-1} \bar{U}_i(t) \varphi_i(\vec{r}). \quad (15)$$

Наша цель – найти эти коэффициенты разложения \bar{U}_i . Введём обозначение:

$$h_F(\bar{U}_h^+, \bar{U}_h^-, \vec{n}) = \left(\vec{\mathbf{F}}, \vec{n} \right)^\Gamma.$$

Это численная потоковая функция, вычисленная на границе элемента ∂K , зависящая от значений приближенного решения по обе стороны границы элемента и от направления единичного нормального вектора \vec{n} . Величины, обозначенные знаком «-», вычисляются на границе ∂K элемента K по значениям внутри элемента K , в то время как величины, обозначенные через «+», вычисляются на границе ∂K по значениям в ячейке, соседней к данной. Воспользовавшись этим обозначением, а также применив разложение (15), мы можем переписать уравнение (14) в виде:

$$\sum_{i=0}^{N-1} \left(\frac{\partial \bar{U}_i}{\partial t} \int_K \varphi_i \varphi_j d\Omega \right) + \oint_{\partial K} \varphi_j h_F(\bar{U}_h^+, \bar{U}_h^-, \vec{n}) d\sigma - \int_K \left(\vec{\nabla} \varphi_j, \vec{\mathbf{F}} \right) d\Omega = 0. \quad (16)$$

Записав последнее уравнение для всех базисных функций φ_j , для каждой ячейки мы получим систему из N линейных уравнений с N неизвестными \bar{U}_i , которую можно записать в следующем виде:

$$\mathbf{A} \cdot \frac{\partial \tilde{U}}{\partial t} = \tilde{R},$$

где \mathbf{A} – матрица масс:

$$\mathbf{A}_{ij} = \int_K \varphi_i \varphi_j d\Omega, \quad (17)$$

а \tilde{R} – вектор правой части:

$$\tilde{R} = \int_K \left(\vec{\nabla} \varphi_j, \vec{\mathbf{F}} \right) d\Omega - \oint_{\partial K} \varphi_j h_F(\bar{U}_h^+, \bar{U}_h^-, \vec{n}) d\sigma.$$

Тогда мы можем записать:

$$\frac{\partial \tilde{U}}{\partial t} = \mathbf{A}^{-1} \cdot \tilde{R}.$$

Для решения последнего уравнения можно использовать различные методы интегрирования по времени, например методы Рунге-Кутты. При вычислении

правой части интегрирование по ячейке и по её границе проводится по Гауссовым точкам.

Далее опишем алгоритм решения более подробно. Для применения разрывного метода Галёркина для решения уравнения Навье-Стокса перепишем уравнение (1) (которое содержит вторые производные по пространственным переменным в случае, если в нём присутствуют вязкий и теплопроводный члены функции потока) в виде системы из трёх уравнений первого порядка:

$$\begin{cases} \hat{\tau} = \mu \left(\vec{\nabla} \otimes \vec{v} + (\vec{\nabla} \otimes \vec{v})^T \right) + \left(\lambda - \frac{2}{3}\mu \right) \hat{\delta}(\vec{\nabla}, \vec{v}), \\ \vec{q} = \hat{k} \cdot \vec{\nabla} T, \\ \partial_t \bar{U} + \left(\vec{\nabla}, \vec{F}^E(U) \right) - \left(\vec{\nabla}, \vec{F}^V(\bar{U}, \hat{\tau}) \right) - \left(\vec{\nabla}, \vec{q} \right) = 0. \end{cases} \quad (18)$$

На каждом шаге по времени эта система будет решаться последовательно. Вначале будут вычисляться вязкая и теплопроводная потоковые функции относительно значений консервативных переменных на предыдущем временном слое (первые два уравнения системы), а затем будут вычисляться значения консервативных переменных на новом временном слое относительно переменных на предыдущем временном слое и этих только что вычисленных потоковых функций.

При вычислении Эйлеровой потоковой функции для вычисления поверхностного интеграла проекция потоковой функции в Гауссовой точке на границе ячейки на вектор внешней нормали к этой границе вычисляется по алгоритму Годунова, Лакса-Фридрихса или HLLC. При этом для вычисления потоковой функции используются значения консервативных переменных как в самой ячейке, так и в соседней (с другой стороны границы).

Для вычисления тензора вязкости $\hat{\tau}$ и градиента температуры \vec{q} их компоненты, так же как и решение исходной системы (1), будем искать приближённо в каждой ячейке в виде разложения на базисные функции (5):

$$\hat{\tau}(\vec{r}, t) = \sum_{i=0}^{N-1} \hat{\tau}_i(t) \varphi_i(\vec{r}), \quad (19)$$

$$\vec{q}(\vec{r}, t) = \sum_{i=0}^{N-1} \vec{q}_i(t) \varphi_i(\vec{r}). \quad (20)$$

Покажем, как вычисляются коэффициенты разложения тензора вязких напряжений. Подставим в левую часть уравнения (3) разложение (19), домножим

полученное уравнение на φ_j и проинтегрируем его по ячейке. Получим:

$$\begin{aligned} & \sum_{i=0}^{N-1} \hat{\tau}_i \int_K \varphi_i \varphi_j d\Omega = \\ & \int_K \mu \varphi_j \left[\vec{\nabla} \otimes \vec{v} + (\vec{\nabla} \otimes \vec{v})^T \right] d\Omega + \\ & \hat{\delta} \int_K \left(\lambda - \frac{2}{3} \mu \right) \varphi_j (\vec{\nabla}, \vec{v}) d\Omega. \end{aligned}$$

Коэффициент динамической вязкости μ является вычисляемой величиной, и в разных точках может принимать разные значения, поэтому выносить его за знак интеграла нельзя. Проинтегрируем оба слагаемых в правой части по частям, получим:

$$\begin{aligned} & \sum_{i=0}^{N-1} \hat{\tau}_i \int_K \varphi_i \varphi_j d\Omega = \\ & \int_K \mu \left[\vec{\nabla} \otimes \varphi_j \vec{v} + (\vec{\nabla} \otimes \varphi_j \vec{v})^T \right] d\Omega - \int_K \mu \left[\vec{\nabla} \varphi_j \otimes \vec{v} + \vec{v} \otimes \vec{\nabla} \varphi_j \right] d\Omega + \\ & \hat{\delta} \left[\int_K \left(\lambda - \frac{2}{3} \mu \right) (\vec{\nabla}, \varphi_j \vec{v}) d\Omega - \int_K \left(\lambda - \frac{2}{3} \mu \right) (\vec{\nabla} \varphi_j, \vec{v}) d\Omega \right]. \end{aligned}$$

Воспользуемся формулой Остроградского-Гаусса (13) и формулой Остроградского-Гаусса для тензорной производной:

$$\int_V (\vec{\nabla} \otimes \vec{f}) d\Omega = \oint_S (\vec{f} \otimes \vec{n})^\Gamma d\sigma,$$

получим:

$$\begin{aligned} & \sum_{i=0}^{N-1} \hat{\tau}_i \int_K \varphi_i \varphi_j d\Omega = \\ & \oint_{\partial K} \varphi_j [\mu (\vec{v} \otimes \vec{n} + \vec{n} \otimes \vec{v})]^\Gamma d\sigma - \int_K \mu \left[\vec{\nabla} \varphi_j \otimes \vec{v} + \vec{v} \otimes \vec{\nabla} \varphi_j \right] d\Omega + \\ & \hat{\delta} \left[\oint_{\partial K} \varphi_j \left(\left(\lambda - \frac{2}{3} \mu \right) (\vec{v}, \vec{n}) \right)^\Gamma d\sigma - \int_K \left(\lambda - \frac{2}{3} \mu \right) (\vec{\nabla} \varphi_j, \vec{v}) d\Omega \right]. \end{aligned}$$

Сгруппируем поверхностные и объёмные интегралы:

$$\begin{aligned}
& \sum_{i=0}^{N-1} \hat{\tau}_i \int_K \varphi_i \varphi_j d\Omega = \\
& \oint_{\partial K} \varphi_j \left[\mu (\vec{v} \otimes \vec{n} + \vec{n} \otimes \vec{v}) + \hat{\delta} \left(\lambda - \frac{2}{3} \mu \right) (\vec{v}, \vec{n}) \right]^\Gamma d\sigma - \\
& \int_K \left[\mu (\vec{\nabla} \varphi_j \otimes \vec{v} + \vec{v} \otimes \vec{\nabla} \varphi_j) + \hat{\delta} \left(\lambda - \frac{2}{3} \mu \right) (\vec{\nabla} \varphi_j, \vec{v}) \right] d\Omega.
\end{aligned} \tag{21}$$

Например, для компоненты тензора τ_{xx} мы получим:

$$\begin{aligned}
& \sum_{i=0}^{N-1} \tau_{xxi} \int_K \varphi_i \varphi_j d\Omega = \\
& \oint_{\partial K} \varphi_j \left[2\mu v_x n_x + \left(\lambda - \frac{2}{3} \mu \right) (\vec{v}, \vec{n}) \right]^\Gamma d\sigma - \\
& \int_K \left[2\mu \frac{\partial \varphi_j}{\partial x} v_x + \left(\lambda - \frac{2}{3} \mu \right) (\vec{\nabla} \varphi_j, \vec{v}) \right] d\Omega,
\end{aligned}$$

а для компоненты тензора τ_{xy}

$$\begin{aligned}
& \sum_{i=0}^{N-1} \tau_{xyi} \int_K \varphi_i \varphi_j d\Omega = \\
& \oint_{\partial K} \varphi_j [\mu (v_x n_y + v_y n_x)]^\Gamma d\sigma - \\
& \int_K \left[\mu \left(\frac{\partial \varphi_j}{\partial y} v_x + \frac{\partial \varphi_j}{\partial x} v_y \right) \right] d\Omega.
\end{aligned}$$

Записав уравнение (21) для всех базисных функций φ_j , мы получим систему из N линейных уравнений с N неизвестными $\hat{\tau}_i$, которую можно записать в следующем виде:

$$\mathbf{A} \hat{\boldsymbol{\tau}} = \hat{\mathbf{R}},$$

где \mathbf{A} – матрица масс (17), а $\hat{\mathbf{R}}$ – вектор правой части, равный разности поверхностного и объёмного интегралов:

$$\hat{\mathbf{R}} = \hat{\mathbf{I}}^{VB} - \hat{\mathbf{I}}^{VV}.$$

Вернёмся теперь к вычислению поверхностного интеграла от функции вязкого потока:

$$\tilde{I}^{VB} = \oint_{\partial K} \tilde{\varphi} \left(\vec{\mathbf{F}}^V, \vec{n} \right)^\Gamma d\sigma.$$

Для его вычисления нужно знать компоненты тензора вязких напряжений в Гауссовых точках на границе ячейки. Покажем, как они вычисляются. Введём обозначения среднего значения $\{g\} = \frac{g^+ + g^-}{2}$ и скачка $[g] = g^+ - g^-$ для величины g произвольного типа. Здесь верхним индексом «-» отмечены значения величин к Гауссовой точке на границе в текущей ячейке, а индексом «+» – в той же точке, но в соседней ячейке. Тогда значение тензора вязких напряжений на границе ячейки задаётся формулой:

$$\hat{\tau}^\Gamma = \{\hat{\tau}\} + C_1 \hat{\delta}([\vec{v}], \vec{n}) + C_2 \frac{[\vec{v}] \otimes \vec{n} + \vec{n} \otimes [\vec{v}]}{2},$$

где C_1 и C_2 – некоторые константы.

Перейдём к описанию потоковой функции теплопроводности. В рассматриваемом изотропном случае вектор \vec{q} определяется как

$$\vec{q} = \kappa \vec{\nabla} T,$$

где κ – коэффициент теплопроводности.

Введём обозначение

$$\vec{G} = \vec{\nabla} T, \quad (22)$$

тогда

$$\vec{q} = \kappa \vec{G}. \quad (23)$$

Так же как и вектор \vec{q} , вектор \vec{G} будем искать в виде разложения

$$\vec{G}(\vec{r}, t) = \sum_{i=0}^{N-1} \vec{G}_i(t) \varphi_i(\vec{r}). \quad (24)$$

Домножим уравнение (22) на φ_j и проинтегрируем по ячейке, подставив в левую часть формулу (24):

$$\sum_{i=0}^{N-1} \vec{G}_i \int_K \varphi_i \varphi_j d\Omega = \int_K \varphi_j \vec{\nabla} T d\Omega.$$

Проинтегрировав правую часть уравнения по частям, получим:

$$\sum_{i=0}^{N-1} \vec{G}_i \int_K \varphi_i \varphi_j d\Omega = \int_K \vec{\nabla} (\varphi_j T) d\Omega - \int_K (\vec{\nabla} \varphi_j) T d\Omega.$$

Воспользуемся формулой Остроградского-Гаусса (13) и в первом слагаемом в правой части перейдём к интегрированию по границе ячейки:

$$\sum_{i=0}^{N-1} \vec{G}_i \int_K \varphi_i \varphi_j d\Omega = \oint_{\partial K} \varphi_j T^\Gamma \vec{n} d\sigma - \int_K (\vec{\nabla} \varphi_j) T d\Omega.$$

Записав последнее уравнение для всех базисных функций φ_j , мы получим для каждой компоненты вектора \vec{G} систему из N линейных уравнений с N неизвестными G_{ij} , которую можно записать в следующем виде:

$$\mathbf{A} \tilde{G}_i = \tilde{R}_i^G,$$

где \mathbf{A} – матрица масс (20), а \tilde{R}_i^G – вектор правой части, равный разности поверхностного и объёмного интегралов:

$$\tilde{R}_i^G = \tilde{I}_i^{GB} - \tilde{I}_i^{GV}.$$

При интегрировании по границе ячейки по Гауссовым точкам берём среднее арифметическое значение T:

$$T^\Gamma = \{T\} = \frac{T^+ + T^-}{2}.$$

Теперь можно вычислить коэффициенты разложения (20) вектора \vec{q} . Домножим уравнение (23) на функцию φ_j и проинтегрируем по ячейке, воспользовавшись разложениями (20) и (24):

$$\sum_{i=0}^{N-1} \vec{q}_i \int_K \varphi_i \varphi_j d\Omega = \sum_{i=0}^{N-1} \vec{G}_i \int_K \kappa \varphi_i \varphi_j d\Omega,$$

или, для каждой компоненты вектора \vec{q} , систему уравнений вида

$$\mathbf{A} \tilde{q}_i = \tilde{R}_i^q,$$

где \mathbf{A} – матрица масс (20), а \tilde{R}_i^q – вектор правой части.

По поводу вычисления коэффициента теплопроводности κ . Коэффициент теплопроводности в общем случае есть некоторая функция от координат, температуры, плотности и т.п. Вид этой зависимости зависит от конкретной задачи.

Вернёмся к вычислению поверхностного интеграла от потоковой функции теплопроводности:

$$\tilde{I}^{TB} = \oint_{\partial K} \tilde{\varphi} \left(\vec{\mathbf{F}}^T, \vec{n} \right)^\Gamma d\sigma.$$

Вектор \vec{q} в Гауссовых точках на границе ячейки, нужный для вычисления этого интеграла, вычисляется по следующей формуле:

$$\vec{q}^T = \{\vec{q}\} - C_3 [T] \vec{n},$$

где C_3 – некоторая неотрицательная константа. Также здесь используются обозначения среднего значения и скачка, введённые выше.

Граничные условия

На границе области в приграничных ячейках потоковые функции определяются по особому правилу, в зависимости от типа краевых условий каждой конкретной задачи. Остановимся на реализации некоторых из них.

Пусть T_k – ячейка, лежащая на границе ∂T , $\vec{n} = (n_x, n_y, n_z)^T$ – единичный вектор внешней нормали к границе, а $\vec{\tau}_1 = (\tau_{1x}, \tau_{1y}, \tau_{1z})^T$ и $\vec{\tau}_2 = (\tau_{2x}, \tau_{2y}, \tau_{2z})^T$ – два каких-либо ортогональных между собой тангенциальных единичных вектора. Эти три вектора образуют новый ортогональный базис. Разложим вектор скорости на границе ячейки по этому базису. Обозначим через V_n нормальную компоненту скорости, а через V_{τ_1} и V_{τ_2} – тангенциальные компоненты. Тогда значения консервативных переменных с другой стороны границы для разных типов граничных условий определяются по следующим правилам.

Граничные условия типа «вход»

Значения численных потоков в квадратурных точках на границе с внешней стороны границы определяются исходя из начальных условий.

Граничные условия типа «выход»

Значения численных потоков в квадратурных точках на границе с внешней стороны границы совпадают со значениями в тех же точках внутри ячейки.

Граничные условия типа «плоскость симметрии» и типа «стенка с проскальзыванием потока».

Значения численных потоков в квадратурных точках на границе определяются из следующих условий:

$$\begin{aligned} \rho_{out}(tsi) &= \rho_{in}(tsi), \\ V_{n_out}(tsi) &= -V_{n_in}(tsi), \\ V_{\tau_1_out}(tsi) &= V_{\tau_1_in}(tsi), \\ V_{\tau_2_out}(tsi) &= V_{\tau_2_in}(tsi), \\ E_{out}(tsi) &= E_{in}(tsi), \end{aligned}$$

т.е. нормальная составляющая скорости меняет знак, а значения всех остальных переменных совпадают со значениями в тех же точках внутри ячейки.

Граничные условия типа «стенка с прилипанием потока»

На стенке с условием прилипания для вычисления градиента скорости, необходимого при вычислении компонент тензора вязких напряжений в

квадратурных точках на границе, будем использовать фактические условия на физические величины. Так как на стенке с прилипанием компоненты скорости равны нулю, а поток, направленный к стенке, должен отразиться от стенки, то тангенциальные производные всех компонент скорости также равны нулю, а производная нормальной скорости вдоль нормали к границе равна своему внутреннему значению. Производная плотности вдоль нормали к границе равна нулю, поскольку из-за прилипания течение стремится к несжимаемому состоянию. Производная температуры вдоль нормали к границе равна нулю в силу отсутствия тепловых потоков в стенку. Тангенциальные производные плотности и температуры берутся из расчетной ячейки внутри области [54, 55]. Из перечисленных условий выражаются условия на градиенты консервативных переменных и, следовательно, определяются градиенты скорости непосредственно в квадратурных точках на границе.

Пусть T_k – ячейка, лежащая на границе ∂T с условием прилипания. Условие прилипания, или no-slip-wall, характеризуется равенством нулю компонент скорости, т.е.

$$\vec{v}|_{\partial T} = (v_x, v_y, v_z)^T \Big|_{\partial T} = \vec{0}.$$

Построим линейную функцию $\rho V_n = \sum_{i=0}^3 (\rho V_n)_i \varphi_i$ по значениям в 4 точках: в t_0 – центре масс ячейки T_k , и в трех точках t_{s1}, t_{s2}, t_{s3} , лежащих на грани ячейки T_k (например, в вершинах грани).

$$\begin{aligned} \rho \vec{v}|_{\partial T} &= \vec{0}, \\ \rho \vec{v}|_{t_0} &= \overline{\rho \vec{v}} = \frac{1}{|T|} \int_{T_k} (\rho \vec{v})_h d\Omega. \end{aligned} \quad (25)$$

Заметим, что формулах (21) интегралы по границе с условием прилипания равны нулю. Этот факт будем учитывать при вычислениях тензора вязкости. Получим:

$$\frac{\partial V_k}{\partial \vec{n}} \Big|_{(t_{si})} = \frac{\partial V_k}{\partial \vec{n}} \Big|_{in(t_{si})}, \quad \frac{\partial V_k}{\partial \vec{\tau}1} \Big|_{(t_{si})} = 0, \quad \frac{\partial V_k}{\partial \vec{\tau}2} \Big|_{(t_{si})} = 0; \quad k = n, \tau 1, \tau 2. \quad (26)$$

Для примера распишем подробно производную по направлению $\frac{\partial V_n}{\partial \vec{n}}$. Для этого воспользуемся тензорной формой записи, позволяющей кратко записывать сложные формулы. В тензорной форме записи по повторяющимся индексам подразумевается суммирование. Например, скалярное произведение векторов в тензорном виде записывается как $(\vec{f}, \vec{g}) = f_i g_i$, а дивергенция вектора – как $\nabla \vec{f} = \frac{\partial f_i}{\partial i}$. Тогда $V_n = (\vec{v}, \vec{n}) = v_i n_i$. Получим:

$$\frac{\partial V_n}{\partial \vec{n}} = \left(\vec{\nabla} V_n, \vec{n} \right) = \frac{\partial V_n}{\partial j} n_j = \frac{\partial v_i}{\partial j} n_i n_j; \quad i, j = x, y, z.$$

Записав аналогичные формулы для всех девяти уравнений (26), мы получим систему из девяти линейных уравнений в девятью неизвестными $\frac{\partial v_i}{\partial j}$ вида

$$AV_0 = V_s, \quad (27)$$

где

$$A = \|k_i m_j\|; k, m = n, \tau 1, \tau 2; i, j = x, y, z; A^{-1} = A^T,$$

$$V_0 = \left(\frac{\partial v_i}{\partial j} \Big|_{(t_{si})} \right)^T - \text{вектор неизвестных,}$$

$V_s = \left(\frac{\partial V_n}{\partial \vec{n}} \Big|_{in(t_{si})}, 0, 0, \frac{\partial V_{\tau 1}}{\partial \vec{n}} \Big|_{in(t_{si})}, 0, 0, \frac{\partial V_{\tau 2}}{\partial \vec{n}} \Big|_{in(t_{si})}, 0, 0 \right)^T$ – правая часть, для вычисления которой используется полиномиальное представление вектора консервативных переменных в ячейке:

$$\frac{\partial V_n}{\partial \vec{n}} = \frac{\frac{\partial(\rho v_i)}{\partial j} \Big|_{in(t_{si})}}{\rho \Big|_{in(t_{si})}} n_i n_j; i, j = x, y, z. \quad (28)$$

Так как $\rho v_i = \sum_{j=0}^{N-1} (\rho v_i)_j \varphi_j$ – полином, то компоненты $\frac{\partial(\rho v_i)}{\partial j}$ можно легко вычислить. Аналогично вычисляются и другие компоненты правой части V_s . Решив систему уравнений (27), получим вектор V_0 , компоненты которого используются для вычисления компонентов тензора вязких напряжений $\hat{\tau}$ по формулам (3).

Так как скорость на границе с условием прилипания равна нулю, то диффузионный поток, с учётом симметричности тензора вязкости, равен

$$\left(\vec{\nabla}, \vec{\mathbf{F}} \right) = \left\{ \begin{array}{c} 0 \\ (\vec{\tau}_x, \vec{n}) \\ (\vec{\tau}_y, \vec{n}) \\ (\vec{\tau}_z, \vec{n}) \\ 0 \end{array} \right\}.$$

Лимитеры

В случае, если решение исходной задачи содержит разрывы или области высоких градиентов, то для обеспечения монотонности необходимо ввести так называемые ограничители наклона $\Lambda \Pi_h$. Такие ограничители для каждой консервативной переменной u_h для любого элемента $T_k \in \Omega$ будем строить, исходя из условия выполнения законов сохранения:

$$\int_{T_k} \Lambda \Pi_h u_h d\Omega = \int_{T_k} u_h d\Omega. \quad (29)$$

Перейдём к описанию лимитера Кокбурна. Разложение одной консервативной переменной u_h в одной ячейке, согласно (15), имеет вид:

$$u_h = \sum_{i=0}^{N-1} u_i \varphi_i. \quad (30)$$

Если степень многочлена этого разложения больше единицы, то вначале это разложение нужно линеаризовать, т.е. представить в виде

$$u_h^l = \sum_{i=0}^D \tilde{u}_i \varphi_i, \quad (31)$$

где D – размерность пространства. Коэффициенты \tilde{u}_i этого линейного разложения будем искать следующим образом. Спроецируем функцию u_h на пространство линейных полиномов. Коэффициенты разложения \tilde{u}_i найдём из условия ортогональности невязки $u_h - u_h^l$ всем линейным многочленам $\varphi_j, j = \overline{0, D}$:

$$\int_{T_k} (u_h - u_h^l) \varphi_j d\Omega = 0, j = \overline{0, D},$$

или, с учётом (31)

$$\int_{T_k} u_h \varphi_j d\Omega = \sum_{i=0}^D \tilde{u}_i \int_{T_k} \varphi_i \varphi_j d\Omega, j = \overline{0, D}. \quad (32)$$

Мы получили систему из $D + 1$ линейных уравнений с $D + 1$ неизвестными \tilde{u}_i , решив которую, мы найдём требуемые коэффициенты линейного разложения.

В случае, если исходное разложение (30) консервативной переменной u_h уже линейное, то полагаем $u_h^l \equiv u_h$.

Сделаем ещё несколько замечаний.

Во-первых, функция φ_0 тождественно равна единице, таким образом, первое уравнение системы (32) выражает закон сохранения (29).

Во-вторых, легко показать, что интеграл по любой ячейке от линейного многочлена $\vec{r} - \vec{r}_c$, где \vec{r}_c – центр масс ячейки, равен нулю. Действительно, по определению центра масс

$$\vec{r}_c = \frac{\int \vec{r} d\Omega}{\int d\Omega},$$

или

$$\int (\vec{r} - \vec{r}_c) d\Omega = \vec{0}.$$

Следовательно, в правой части первого уравнения системы (32) для функции φ_0 все слагаемые, кроме первого, обращаются в ноль, а первое равно объёму

ячейки, умноженному на \tilde{u}_0 . В остальных уравнениях в ноль обращается первое слагаемое. Система (32) может быть записана в следующем упрощённом виде:

$$\begin{cases} \tilde{u}_0 = \frac{1}{|T_k|} \int_{T_k} u_h d\Omega, \\ \int_{T_k} u_h \varphi_j d\Omega = \sum_{i=1}^D \tilde{u}_i \int_{T_k} \varphi_i \varphi_j d\Omega, j = \overline{1, D}. \end{cases} \quad (33)$$

Коэффициент \tilde{u}_0 выражает т.н. интегральное среднее значение консервативной переменной u_h в ячейке. Требование выполнения законов сохранения (29) эквивалентно требованию сохранения интегрального среднего по ячейке. Из равенства нулю интеграла по ячейке от линейного многочлена $\vec{r} - \vec{r}_c$ следует, во-первых, что вычисленное по первому уравнению системы (33) значение коэффициента \tilde{u}_0 окончательное и в процессе лимитирования меняться не должно, а во-вторых, что изменение значений других коэффициентов $\tilde{u}_i, i = \overline{1, D}$ интегральное среднее не меняет. В процессе лимитирования мы можем менять эти коэффициенты, причем в сторону уменьшения по модулю.

В данной работе мы будем описывать лимитирование только для ячеек, у которых геометрический центр (среднее арифметическое координат вершин) совпадает с центром масс. Для треугольников и тетраэдров данное условие выполнено всегда. Для других типов ячеек эти два центра могут отличаться (хотя могут и совпадать).

Обозначим через T_0 рабочую ячейку, в которой необходимо провести ограничение функций (лимитирование), а через T_i – ячейки, имеющие общую грань с ячейкой $T_0, i = \overline{1, N_g}$, где N_g – число граней ячейки $T_0, N_g \geq D + 1$. Далее введём обозначения:

\vec{t}_0 – геометрический центр рабочей ячейки T_0 ;

\vec{t}_i – центр масс ячейки T_i ;

\vec{t}_{0i} – геометрический центр грани с номером i .

Построим вектора:

$\vec{l}_{0i} = \vec{t}_{0i} - \vec{t}_0$;

$\vec{l}_j = \vec{t}_j - \vec{t}_0$.

Для каждого вектора \vec{l}_{0i} найдём угол (обычный или трёхгранный в зависимости от размерности пространства), образованный теми D векторами \vec{l}_j , в котором лежит вектор \vec{l}_{0i} , и найдём его разложение по этим векторам:

$$\vec{l}_{0i} = \sum_{j=1}^D \alpha_{ij} \vec{l}_j, i = \overline{1, N_g}. \quad (34)$$

Алгоритм выбора требуемых векторов \vec{l}_{ij} зависит от типа ячейки и будет описан ниже. Далее обозначим:

$u_{0i} = u_h^l(\vec{t}_{0i})$ – значение функции u_h^l в геометрическом центре грани с номером i ;

$\Delta u_{0i} = u_{0i} - \tilde{u}_0$ – разность между значением функции u_h^l в геометрическом центре грани с номером i и интегральным средним в рабочей ячейке.

Вычислим интегральные средние значения функции u_h в соседних ячейках:

$$\tilde{u}_0^{(i)} = \frac{1}{|T_i|} \int_{T_i} u_h d\Omega, i = \overline{1, N_g}. \quad (35)$$

Введём обозначение

$$\Delta \tilde{u}_{0i} = \sum_{j=1}^D \alpha_{i_j} \left(\tilde{u}_0^{(i_j)} - \tilde{u}_0 \right), i = \overline{1, N_g}, \quad (36)$$

где α_{i_j} – коэффициенты из разложения (34) вектора \vec{l}_{0i} . При этом для каждой грани в формулу для $\Delta \tilde{u}_{0i}$ входят интегральные средние только тех ячеек, которые вошли в разложение (34). Геометрический смысл величины $\Delta \tilde{u}_{0i}$ следующий. В $D+1$ -мерном пространстве (\vec{r}, u_h) для каждого $i = \overline{1, N_g}$ мы проводим D -мерную гиперплоскость через точки $(\vec{t}_{i_j}, \tilde{u}_0^{(i_j)})$ для ячеек, вошедших в разложение (34), и точку (\vec{t}_0, \tilde{u}_0) рабочей ячейки. Эта гиперплоскость пересекает прямую $\vec{r} = \vec{t}_{0i}$ в некоторой точке (значение линейной функции, задаваемой этой гиперплоскостью, в точке \vec{t}_{0i}). Величина $\Delta \tilde{u}_{0i}$ задаёт разность между этим значением и интегральным средним рабочей ячейки. Сказанное продемонстрируем для одномерного случая на рис. 1.

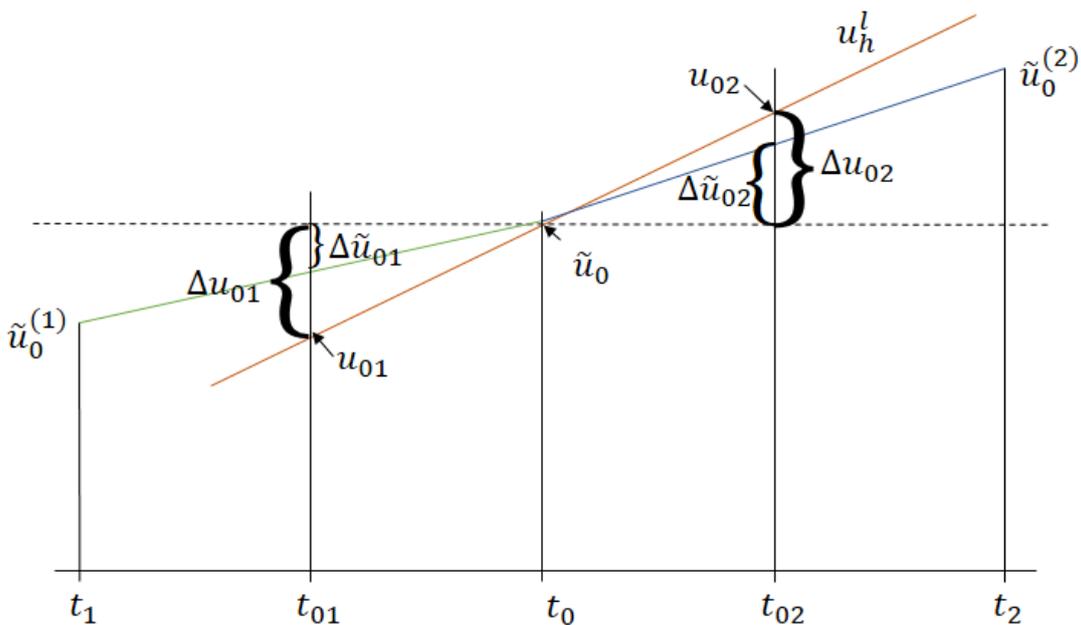


Рис. 1. Величины в одномерном случае

В $D + 1$ -мерном пространстве (\vec{r}, u_h) D -мерная гиперплоскость h_h^l проходит (по определению u_{0i}) через все точки вида $(\vec{t}_{0i}, u_{0i}), i = \overline{1, N_g}$, лежащие над геометрическими центрами граней. Среди этих точек некоторым образом (в зависимости от типа ячейки) выберем $D + 1$ точку (например, для треугольников и тетраэдров берутся все точки). Номера выбранных граней обозначим через $k_i, i = \overline{0, D}$. Тогда линейную функцию u_h^l можно представить в виде

$$u_h^l = \sum_{i=0}^D u_{0k_i} \psi_i = \tilde{u}_0 + \sum_{i=0}^D \Delta u_{0k_i} \psi_i, \quad (37)$$

где функция

$$\psi_i = \begin{cases} 1 & \text{в } \vec{t}_{0k_i}; \\ 0 & \text{в } \vec{t}_{0k_j}, j \neq i. \end{cases}$$

Второе равенство в уравнении (37) следует из очевидного тождества:

$$\sum_{i=0}^D \psi_i \equiv 1.$$

Вычислим величины

$$\Delta u_{0i}^* = \bar{m}(\Delta u_{0i}, \nu \Delta \tilde{u}_{0i}),$$

где $\nu \geq 1$, а \bar{m} – функция «TVB minmod» (в случае TVB-ограничителя):

$$\bar{m}(a_1, a_2) = \begin{cases} a_1, & \text{если } |a_1| \leq M \cdot (\Delta x)^2, \\ m(a_1, a_2) & \text{иначе,} \end{cases}$$

где $M \geq 0$ – заданная константа, Δx – характерный размер сетки, $m(a_1, a_2)$ – TVD-ограничитель:

$$m(a_1, a_2) = \begin{cases} s \cdot \min(|a_1|, |a_2|), & \text{если } a_1 \text{ и } a_2 \text{ одного знака, } s = \text{sign}(a_1), \\ 0 & \text{иначе.} \end{cases}$$

Далее для вычисленных величин Δu_{0i}^* проверяется выполнение некоторого условия (для каждого типа ячейки своего, например, для треугольников и тетраэдров это условие выглядит как $\sum_{i=1}^{D+1} \Delta u_{0i}^* = 0$). Если это условие не выполнено, то величины Δu_{0i}^* подвергаются коррекции в сторону уменьшения, алгоритм коррекции для каждого типа ячеек свой:

$$\Delta \tilde{u}_{0i}^* = \text{correct}(\Delta u_{0i}^*). \quad (38)$$

Если условие выполнено, то полагаем $\Delta \tilde{u}_{0i}^* = \Delta u_{0i}^*$.
Окончательно полагаем

$$\hat{u}_h = \tilde{u}_0 + \sum_{i=0}^D \Delta \tilde{u}_{0k_i}^* \psi_i.$$

Если после этого отлимитированная функция не изменится: $\hat{u}_h = u_h^l$, то ограничитель полагаем равным тождественному оператору

$$\Lambda \Pi_h u_h = u_h,$$

в противном случае

$$\Lambda \Pi_h u_h = \hat{u}_h.$$

Рассмотрим особенности алгоритма лимитирования для разных типов ячеек. Нижним индексом $_{0i}$ будем обозначать грань, противоположную грани $_{0i}$.

1. Треугольник и тетраэдр

а) Выбор векторов для разложения (34). В двумерном случае (треугольник или четырёхугольник) всё просто. Вектора \vec{l}_i разбивают плоскость на 3 (или 4) угла. Для каждого вектора \vec{l}_{0i} нужно найти тот из них, в котором он лежит. В случае тетраэдра всё немного сложнее. Рассмотрим все четыре возможных тройки векторов \vec{l}_i . Каждая такая тройка образует трёхгранный телесный угол, разбивающий пространство на две части. В качестве телесного угла, образованного данной тройкой векторов, примем ту часть пространства, в которой не лежит оставшийся вектор. Тогда эти четыре тройки векторов разобьют всё пространство на четыре непересекающихся трёхгранных телесных угла. Для каждого вектора \vec{l}_{0i} нужно найти тот из них, в котором он лежит, и для разложения взять тройку векторов, образующих выбранный телесный угол.

б) Выбор граней для разложения (37). Как уже говорилось выше, для треугольников и тетраэдров берутся все грани.

с) Условие для величин Δu_{0i}^* имеет вид

$$\sum_{i=1}^{D+1} \Delta u_{0i}^* = 0.$$

д) Алгоритм корректировки (38). Вычисляем суммарный вклад положительных и отрицательных членов Δu_{0i}^* :

$$\begin{aligned} pos &= \sum_{i=1}^{D+1} \max(0, \Delta u_{0i}^*), \\ neg &= \sum_{i=1}^{D+1} \max(0, -\Delta u_{0i}^*). \end{aligned}$$

Полагаем

$$\theta^+ = \min \left(1, \frac{neg}{pos} \right),$$

$$\theta^- = \min \left(1, \frac{pos}{neg} \right).$$

И, наконец,

$$\Delta \tilde{u}_{0i}^* = \theta^+ \max(0, \Delta u_{0i}^*) - \theta^- \max(0, -\Delta u_{0i}^*).$$

2. Четырёхугольник

а) Выбор векторов для разложения (34). Алгоритм тот же, что и для треугольников.

б) Выбор граней для разложения (37). Выбираются любые три грани.

в) Условие для величин Δu_{0i}^* . Нулю должны быть равны суммы для противоположных граней. Если противоположными являются пары граней (1, 2) и (3, 4), то условие выглядит так:

$$\Delta u_{01}^* + \Delta u_{02}^* = 0,$$

$$\Delta u_{03}^* + \Delta u_{04}^* = 0.$$

д) Алгоритм корректировки (38) следующий:

$$\Delta \tilde{u}_{0i}^* = m(\Delta u_{0i}^*, -\Delta u_{0i-}^*).$$

3. Четырёхугольная пирамида

Пусть основанию пирамиды соответствует грань номер 5, а противоположными являются пары граней (1, 2) и (3, 4).

а) Выбор векторов для разложения (34). Вектора \vec{l}_{0i} раскладываются по одному из 8 трёхгранных телесных углов, образованных всеми возможными тройками векторов (всего их $C_5^3 = 10$), кроме углов, образованных векторами, соединяющими противоположные грани и основание.

б) Выбор граней для разложения (37). Выбираются основание и любые три боковых грани.

в) Условие для величин Δu_{0i}^* выглядит так:

$$1,5\Delta u_{01}^* + 1,5\Delta u_{02}^* + \Delta u_{05}^* = 0,$$

$$1,5\Delta u_{03}^* + 1,5\Delta u_{04}^* + \Delta u_{05}^* = 0.$$

д) Алгоритм корректировки (38). Вычисляем суммарный вклад положительных и отрицательных членов Δu_{0i}^* для тройки, для которой необходимо уменьшить Δu_{05}^* :

$$pos = \sum_i \max(0, \alpha_i \Delta u_{0i}^*),$$

$$neg = \sum_i \max(0, -\alpha_i \Delta u_{0i}^*).$$

Полагаем

$$\theta^+ = \min \left(1, \frac{neg}{pos} \right),$$

$$\theta^- = \min \left(1, \frac{pos}{neg} \right).$$

И, наконец,

$$\Delta \tilde{u}_{0i}^* = \theta^+ \max(0, \Delta u_{0i}^*) - \theta^- \max(0, -\Delta u_{0i}^*).$$

4. Треугольная призма

Пусть треугольным основаниям призмы соответствуют грани номер 1 и 2, а боковым сторонам – грани номер 3, 4 и 5.

а) Выбор векторов для разложения (34). Вектора \vec{l}_{0i} раскладываются по одному из 6 трёхгранных телесных углов, образованных всеми возможными тройками векторов, образованных векторами, соединяющими одно из оснований и две боковые грани.

б) Выбор граней для разложения (37). Выбираются два основания и любые две боковых грани.

с) Условие для величин Δu_{0i}^* выглядит так:

$$\Delta u_{01}^* + \Delta u_{02}^* = 0,$$

$$\Delta u_{03}^* + \Delta u_{04}^* + \Delta u_{05}^* = 0.$$

д) Алгоритм корректировки (38). Если нарушено первое условие, то полагаем

$$\Delta \tilde{u}_{0i}^* = m(\Delta u_{0i}^*, -\Delta u_{0i}^*), i = 1, 2.$$

В случае если нарушено второе условие, то для граней 3, 4 и 5 вычисляем суммарный вклад положительных и отрицательных членов Δu_{0i}^* :

$$pos = \sum_{i=3,4,5} \max(0, \Delta u_{0i}^*),$$

$$neg = \sum_{i=3,4,5} \max(0, -\Delta u_{0i}^*).$$

Полагаем

$$\theta^+ = \min \left(1, \frac{neg}{pos} \right),$$

$$\theta^- = \min \left(1, \frac{pos}{neg} \right).$$

И, наконец,

$$\Delta \tilde{u}_{0i}^* = \theta^+ \max(0, \Delta u_{0i}^*) - \theta^- \max(0, -\Delta u_{0i}^*), i = 3, 4, 5.$$

5. Гексаэдр (четырёхугольная призма)

Пусть противоположными являются пары граней (1, 2), (3, 4) и (5, 6).

а) Выбор векторов для разложения (34). Вектора \vec{l}_{0i} раскладываются по одному из 8 трёхгранных телесных углов, образованных всеми возможными тройками векторов, кроме углов, образованных векторами, соединяющими противоположные грани.

б) Выбор граней для разложения (37). Выбираются три любых грани, имеющие общую вершину, и одна любая грань из трёх оставшихся.

с) Условие для величин Δu_{0i}^* выглядит так:

$$\Delta u_{01}^* + \Delta u_{02}^* = 0,$$

$$\Delta u_{03}^* + \Delta u_{04}^* = 0,$$

$$\Delta u_{05}^* + \Delta u_{06}^* = 0.$$

д) Алгоритм корректировки (38) следующий:

$$\Delta \tilde{u}_{0i}^* = m(\Delta u_{0i}^*, -\Delta u_{0i}^*).$$

Библиографический список

1. Cockburn B., Shu C-W. The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems // J. Comput. Phys., 141, 1998, pp. 199-224.
2. Reed W.H., Hill T.R. Triangular mesh methods for the neutron transport equation // Los Alamos Scientific Laboratory Report LA-UR-73-79, 1973 United States. URL: <https://www.osti.gov/servlets/purl/4491151>.
3. Proceedings ECCOMAS Thematic Conference: European Conference on High Order Nonlinear Numerical Methods for Evolutionary PDEs: Theory and Applications HONOM 2017 March 27 - March 31, 2017. URL: https://honom2017.iag.uni-stuttgart.de/wp-content/uploads/2017/03/HONOM_Proceedings_Version2.pdf.
4. Ethan J. Kubatko, Joannes J. Westerink, Clint Dawson. hp Discontinuous Galerkin methods for advection dominated problems in shallow water flow // J. Comput. Methods Appl. Mech. Engrg. 196, 2006, pp. 437–451.
5. Волков А.В. Особенности применения метода Галёркина к решению пространственных уравнений Навье-Стокса на неструктурированных гексаэдральных сетках // Ученые записки ЦАГИ №6, т. XL, 2009.

6. Nastase C.R., Mavriplis D.J. High-order discontinuous Galerkin methods using an hp-multigrid approach // J. Comput. Phys., vol. 213, 2006, pp. 330-357.
7. Luo H., Baum J.D., Löhner R. Fast p-multigrid discontinuous Galerkin method for compressible flow at all speeds // AIAA J., vol. 46, 2008, pp. 635-652.
8. Arnold D.N., Brezzi F., Cockburn B., Marini L.D. Unified analysis of discontinuous Galerkin methods for elliptic problems // SIAM Journal on Numerical Analysis, 2002, 29, pp. 1749–1779.
9. Мысовских И.П. Интерполяционные кубатурные формулы. М.: Наука, 1981.
10. Krasnov M.M., Kuchugov P.A., Ladonkina M.E., Lutsky A.E., Tishkin V.F. Numerical solution of the Navier-Stokes equations by discontinuous Galerkin method // Journal of Physics Conference Series, Vol. 815, №1, 2017.
11. Bassi F., Rebay S. Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier-Stokes equations // Int. J. Numer. Math. Fluids, vol. 40, 2002, pp. 197-207.
12. Краснов М.М., Кучугов П.А., Ладонкина М.Е., Тишкин В.Ф. Разрывный метод Галёркина на трехмерных тетраэдральных сетках. Использование операторного метода программирования // Математическое моделирование, 2017, т. 29, №2, с. 3-22.
13. Краснов М.М., Ладонкина М.Е. Разрывный метод Галёркина на трёхмерных тетраэдральных сетках. Применение шаблонного метапрограммирования языка C++ // Программирование, 2017, №3, с. 41-53.
14. Yasue K., Furudate M., Ohnishi N., Sawada K. Implicit discontinuous Galerkin method for RANS simulation utilizing pointwise relaxation algorithm // Communications in Computational Physics, Vol. 7, №3, 2010, pp. 510–533.
15. Cockburn B. An Introduction to the Discontinuous Galerkin Method for Convection - Dominated Problems, Advanced Numerical Approximation of Nonlinear Hyperbolic Equations // Lecture Notes in Mathematics, 1998, Vol. 1697, pp. 151-268.
16. Krivodonova L. Limiters for high-order discontinuous Galerkin methods // Journal of Computational Physics, 226:1, 2007, pp. 276–296.
17. Krivodonova L., Xin J., Remacle J.-F., Chevogeon N., Flaherty J. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws // Appl. Numer. Math., Vol. 48, №3, 2004, pp. 323-338.

18. Cockburn B., Shu C.W.: Runge-Kutta discontinuous Galerkin methods for convection-dominated problem // *Scientific Comput.* 16, 2001, pp. 173-261.
19. Qiu J., Shu C.-W.: Runge-Kutta Discontinuous Galerkin Method Using WENO Limiters // *SIAM J. Sci. Comput.*, 26(3), 907929. 2006.
20. Shu C.-W. High order WENO and DG methods for time-dependent convection-dominated PDEs: A brief survey of several recent developments // *Journal of Computational Physics*. Vol. 316, 2016, pp. 598-613.
21. Luo H., Baum J.D., Löhner R.: A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids // *Journal Journal of Computational Physics*, Volume 225 Issue 1, 2007, pp. 686-713.
22. Zhu J., Zhong X., Shu C.-W., Qiu J. Runge-Kutta discontinuous Galerkin method using a new type of WENO limiters on unstructured meshes // *Journal of Computational Physics*. Vol. 248, 2013, pp. 200-220.
23. Dumbser M., Balsara D.S., Toro E.F., Munz C.-D. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes // *Journal of Computational Physics*, 227:82098253, 2008.
24. Michael Dumbser. Arbitrary high order PNP schemes on unstructured meshes for the compressible Navier–Stokes equations // *Computers & Fluids* Volume 39, Issue 1, January 2010, pp. 60–76.
25. Persson P.O., Peraire J. Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations // *SIAM J. Sci. Comput.* 30(6), 27092733, 2008.
26. Peraire J., Persson P.-O. Adaptive High-Order Methods in Computational Fluid Dynamics, vol. 2 of *Advances in CFD*, chap. 5 - High-Order Discontinuous Galerkin Methods for CFD, World Scientific Publishing Co., 2011.
27. Волков А.В., Ляпунов С.В. Монотонизация метода конечного элемента в задачах газовой динамики // *Ученые записки ЦАГИ*, т. XL, №4, 2009, с. 15-27.
28. Ладонкина М.Е., Неклюдова О.А., Тишкин В.Ф. Построение лимитера для разрывного метода Галёркина на основе усреднения решения // *Матем. моделирование*, 2018, т. 30, №5, с. 99–116.
29. Haga T., Sawada K. An improved slope limiter for high-order spectral volume methods solving the 3D compressible Euler equations, 2009.
30. Cockburn B., Shu C.-W. The local discontinuous Galerkin method for time dependent convection diffusion system // *SIAM J. Numer. Anal.* 1998. vol. 35, №6, pp. 2440-2463.

31. Arnold D.N., Brezzi F., Cockburn B., Marini D. Discontinuous Galerkin methods for elliptic problems // *Discontinuous Galerkin Methods*, vol. 11 of *Lecture Notes in Computational Science and Engineering*, pp. 89–101, Springer, Berlin, Germany, 2000.
32. Bassi F., Rebay S. Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier-Stokes equations // *Int. J. Numer. Meth. Fluids*, 40, 2002, pp. 197-207.
33. Власенко В.В., Волков А.В., Трошин А.И. Выбор метода аппроксимации вязких членов в методе Галёркина с разрывными базисными функциями // *Ученые записки ЦАГИ*, т. XLIV, №3, 2013.
34. Gottlieb S., Shu C.-W., Tadmor E. Strong stability-preserving high-order time discretization methods // *SIAM Review*. 2001, vol. 43, №1. pp. 89–112.
35. Spiteri Raymond J., Ruuth Steven J. A New Class of Optimal High-Order Strong-Stability-Preserving Time Discretization Methods // *SIAM J. NUMER. ANAL.* 2002. 40, №2, pp. 469–491.
36. Bassi F., Rebay S. A High Order Discontinuous Galerkin Method for Compressible Turbulent Flow // B. Cockburn, G. E. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Method: Theory, Computations and Applications*, *Lecture Notes in Computational Science and Engineering* 11, Springer-Verlag, 2000, pp. 113-123.
37. Rasetarinera P., Hussaini M.Y. An efficient implicit discontinuous spectral Galerkin method // *J. Comput. Phys.*, 172, 2001, pp. 718-738.
38. Hartmann R. Adaptive discontinuous Galerkin methods with shock-capturing for the compressible Navier-Stokes equations // *Int. J. Numer. Meth. Fluids*, 51, 2006, pp. 1131-1156.
39. Hartmann R., Houston P. Symmetric interior penalty DG methods for the compressible Navier-Stokes equations I: Method formulation // *Int. J. Numer. Anal. Model.*, 3, 2006, pp. 1-20.
40. Hartmann R., Houston P. Symmetric interior penalty DG methods for the compressible Navier-Stokes equations II: Goal-oriented a posteriori error estimation // *Int. J. Numer. Anal. Model.*, 3, 2006, pp. 141-162.
41. Dolejší V., Feistauer M. A semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow // *J. Comput. Phys.*, 198, 2004, 727746.

42. Dolejší V. Semi-implicit interior penalty discontinuous Galerkin methods for viscous compressible flows // *Commun. Comput. Phys.*, 4, 2008, pp. 231-274.
43. Yasue K., Furudate M., Ohnishi N., Sawada K. Implicit discontinuous Galerkin method for RANS simulation utilizing pointwise relaxation algorithm // *Communications in Computational Physics*, Vol.7, №3, pp. 510–533, 2010.
44. Sawada K., Yasue K. A linear stability analysis of the cell wise relaxation implicit discontinuous Galerkin method for wave propagation // *Fluid Dynamics Research*, vol. 43, №4, Article ID 041402, 18 pages, 2011.
45. Bassi F., Botti L., Colombo A., Massa F. An implicit high-order discontinuous Galerkin method for incompressible variable density flows // *Proceedings ECCOMAS Thematic Conference: European Conference on High Order Nonlinear Numerical Methods for Evolutionary PDEs: Theory and Applications HONOM 2017 March 27 - March 31, 2017*.
46. Klockner A., Warburton T., Hesthaven J.S.: Nodal discontinuous Galerkin methods on graphics processors // *Journal of Computational Physics*, 228(21):7863-7882, 2009.
47. Lou J., Xia Y., Luo L., Luo H., Edwards J.R., Mueller F. OpenACC-based GPU Acceleration of a p-multigrid Discontinuous Galerkin Method for Compressible Flows on 3D Unstructured Grids // *53rd AIAA Aerospace Sciences Meeting, AIAA SciTech Forum*, (AIAA 2015-0822).
48. Chan J., Wang Z., Modave A., Remacle J.-F., Warburton T. GPU-accelerated discontinuous Galerkin methods on hybrid meshes // *Journal of Computational Physics*, vol. 318, 2015.
49. Краснов М.М. Операторная библиотека для решения трехмерных сеточных задач математической физики с использованием графических плат с архитектурой CUDA // *Математическое моделирование*, 2015, т. 27, №3, с. 109-120.
50. Краснов М.М. Сеточно-операторный подход к программированию задач математической физики // *Диссертация на соискание ученой степени кандидата физико-математических наук*, Москва, 2017. URL: <http://keldysh.ru/council/1/2017-krasnov/avtoref.pdf>
51. Ладонкина М.Е., Неклюдова О.А., Тишкин В.Ф. Использование разрывного метода Галёркина при решении задач гидродинамики // *Мат. моделирование*, 2014, т. 26, №1, с. 17-32.

52. Краснов М.М. Применение символьного дифференцирования для решения ряда вычислительных задач // Препринты ИПМ им.М.В.Келдыша. 2017. №4. 24 с. doi:10.20948/prepr-2017-4. URL: http://keldysh.ru/papers/2017/prep2017_4.pdf.
53. Jameson A., Yoon S. Lower-upper implicit schemes with multiple grids for the Euler equations // AIAA J., vol. 25, 1987, pp. 929-935.
54. Босняков С.М., Михайлов С.В., Подаруев В.Ю., Трошин А.И. Нестационарный разрывный метод Галеркина высокого порядка точности для моделирования турбулентных течений // Матем. моделирование, 2018, т. 30, №5, с. 37-56.
55. Подаруев В.Ю. Метод высокого порядка точности для расчета на суперкомпьютере характеристик турбулентных струй, истекающих из сопел гражданских самолетов // Диссертация на соискание ученой степени кандидата технических наук, Жуковский, 2017.