

УДК 532.529

РЕАЛИЗАЦИЯ ЛАГРАНЖЕВОГО ПОДХОДА К ОПИСАНИЮ ТЕЧЕНИЙ ГАЗА С ЧАСТИЦАМИ НА НЕСТРУКТУРИРОВАННЫХ СЕТКАХ

К. Н. Волков¹, В. Н. Емельянов¹

Рассматриваются вопросы, связанные с численной реализацией лагранжевого подхода к моделированию течений газа с частицами на неструктурированных сетках. Обсуждаются реализации ряда алгоритмов локализации частицы и поиска контрольного объема неструктурированной сетки, в котором находится частица в данный момент времени. Приводятся оценки производительности некоторых алгоритмов.

Ключевые слова: двухфазные течения газа, дискретно-траекторный подход, неструктурированные сетки, метод контрольных объемов, метод частиц.

1. Введение. В расчетах двухфазных течений на основе дискретно-траекторного (лагранжевого) подхода для имитации движения примеси приходится осуществлять массовые расчеты траекторий пробных частиц [1, 2]. Достаточно часто описание движения примеси рассматривается как самостоятельная задача, в которой решаются вопросы эволюции частиц в известном газодинамическом поле (влияние частиц на течение газа не учитывается).

Эффективная реализация вычислений требует разработки как конечно-разностных схем, учитывающих особенности задачи (например, размер частицы, действие на частицу различных силовых факторов, наличие рециркуляционных зон и др.), так и различных вспомогательных алгоритмов, связанных с локализацией частицы в контрольном объеме неструктурированной сетки, поиском ячейки расчетной сетки, в которой находится частица, и интерполяцией газодинамических параметров несущего потока (рис. 1). Использование разностных схем, учитывающих особенности движения частиц мелких и крупных фракций, позволяет сократить время счета и получить выигрыш в характеристиках точности. В основе разностных схем, разработанных в [3], лежит линеаризация уравнений, реализующая принцип замораживания отдельных членов или частей уравнений, а также их приближенное представление в виде упрощенных функциональных зависимостей с последующим аналитическим интегрированием приближенного уравнения на каждом временном шаге. Мелкие частицы следуют вдоль линий тока, поэтому метод расчета таких частиц строится как возмущение соответствующих параметров транспортирующего газа. Для расчета движения крупных частиц используется двухшаговая вычислительная процедура типа предиктор-корректор.

Помимо разностных схем, для реализации лагранжевого подхода существенное значение имеют нахождение контрольного объема неструктурированной сетки, в котором находится частица, и восполнение параметров несущего газа [4]. В то время как эти вопросы сравнительно просто решаются на структурированных сетках (в том числе и криволинейных), их решение на неструктурированных сетках требует более детального рассмотрения и исследования.

2. Основные уравнения. Рассмотрим движение пробной частицы сферической формы под действием силы сопротивления. Уравнение движения частицы в газодинамическом поле, скорость которого v

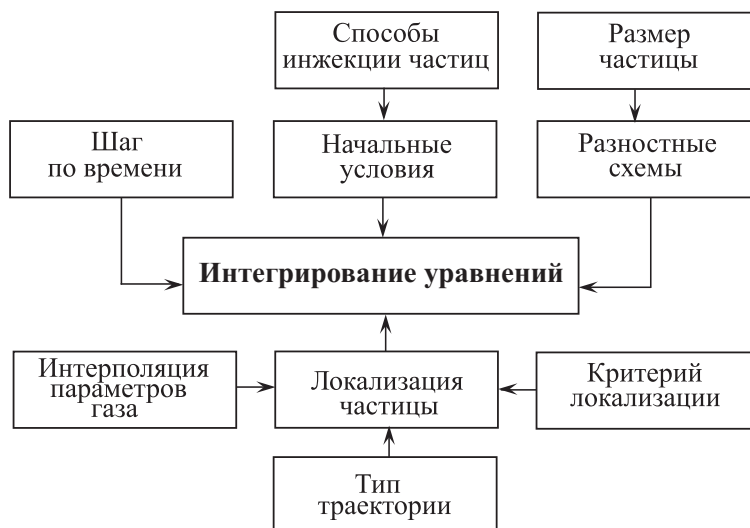


Рис. 1. Реализация дискретно-траекторного подхода

¹ Балтийский государственный технический университет “Военмех” им. Д. Ф. Устинова, физико-механический факультет, 1-я Красноармейская ул., д. 1, 190005, Санкт-Петербург; e-mail: dsci@mail.ru

полагается известной, имеет вид

$$\frac{d\mathbf{v}_p}{dt} = \frac{1}{\tau_p} (\mathbf{v} - \mathbf{v}_p) + \mathbf{a}_p, \quad (1)$$

где \mathbf{a}_p — ускорение, приобретаемое частицей под действием различных сил, исключая силу сопротивления, и τ_p — время релаксации.

Радиус-вектор центра масс частицы находится из уравнения

$$\frac{d\mathbf{r}_p}{dt} = \mathbf{v}_p. \quad (2)$$

Температура частицы влияет на ее движение через поправку к коэффициенту сопротивления. Во многих режимах течения такая поправка мала и не учитывается.

Уравнения (1) и (2) интегрируются вдоль траектории пробной частицы и требуют задания только начальных условий в момент времени $t = 0$.

Скорость газа вычисляется в точках, лежащих на траектории частицы $\mathbf{v} = \mathbf{v}(\mathbf{r}_p)$. Скорость газа находится из решения уравнений Навье–Стокса в контрольных объемах эйлеровой сетки, узлы которой не совпадают с положением частицы. Это приводит к необходимости восполнения (интерполяции) газодинамических параметров несущего потока, что требует локализации частицы и нахождения ячейки или контрольного объема расчетной сетки, в котором находится частица в данный момент времени.

3. Особенности реализации. При реализации лагранжевого подхода к описанию движения дисперсной фазы требуется решить вопросы, связанные с постановкой начальных условий и инжекцией частиц, расчетом шага по времени и предотвращением ряда исключительных ситуаций.

3.1. Способы инжекции частиц. В каждой точке, из которой инжектируется частица, задаются тип частицы (инертная, горячая или какая-либо другая), материал и начальные условия. На практике используются различные способы ввода частиц в расчетную область (рис. 2).

Для исследования движения индивидуальной частицы задаются ее положение и скорость в начальный момент времени (рис. 2 а).

При изучении движения группы частиц задаются координаты начальной и конечной частицы, закон изменения их скорости и число частиц N (рис. 2 б). Параметр ϕ_i , соответствующий частице i в группе, находится из соотношения $\phi_i = \phi_1 + \frac{\phi_N - \phi_1}{N - 1} (i - 1)$. Значения ϕ_1 и ϕ_N соответствуют начальной и конечной частице группы.

При исследовании движения полидисперсной системы частиц определяется распределение частиц по размерам. Обычно задается массовый расход частиц каждой фракции, а также минимальный, максимальный и средний радиусы частиц. Массовый расход частиц каждой фракции связан с размером частиц при помощи экспоненциальной зависимости (Rosin–Rammler law). В случае решения нестационарной задачи определяется время, в течение которого происходит инжекция частиц.

В ряде случаев используется точечный источник частиц (рис. 2 в). Частицы инжектируются из одной точки с различными скоростями.

В трехмерном случае при решении некоторых задач (например, при моделировании движения частиц в струе) находит применение коническое распределение частиц (рис. 2 г) или инжекция частиц с поверхности, ограничивающей расчетную область (например, при расчете течений в каналах со вдувом).

Для определения положения всех частиц в случае конического распределения задаются положение частицы на оси симметрии и ее скорость, радиус основания r и угол полураствора конуса θ . При необходимости задается скорость частиц в окружном направлении и их массовый расход (при учете влияния частиц на газ).

Моделирование движения примеси на основе стохастического варианта дискретно-траекторного подхода, реализации которого обсуждаются в [1, 2], требует задания числа частиц n , выходящих из каждой точки входного сечения с одной и той же средней скоростью, но с различными пульсационными скоростями. Значение $n = 1$ соответствует детерминистическому подходу, основанному на использовании среднего поля течения. Для получения средней картины движения примеси проводится осреднение по ансамблю частиц. Малые значения n достаточно часто приводят к расходимости итерационного процесса.

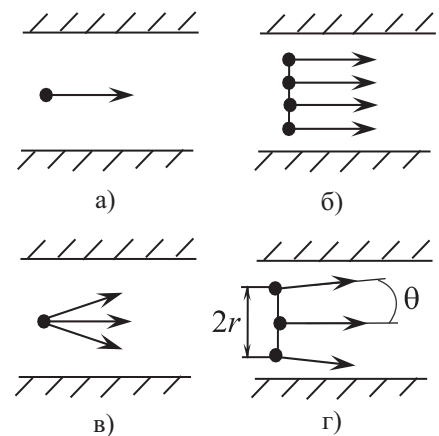


Рис. 2. Способы инжекции частиц

При моделировании движения частиц в области, обладающей свойством повторяемости по одной из пространственных координат, используется условие периодичности. Частицы, покидающие расчетную область, инжектируются с противоположной границы со скоростью, которую они имеют на выходе.

3.2. Виды траекторий. В зависимости от граничных условий и параметров задачи различают следующие типы частиц и их траекторий.

1. Частицы, расчет траекторий которых прерывается из-за того, что они попадают на границу области с граничными условиями исчезновения частиц (trapped trajectories).

2. Частицы, расчет траекторий которых заканчивается после достижения максимального числа шагов по времени (incomplete trajectories).

3. Частицы, расчет траекторий которых заканчивается, поскольку частицы исчезают внутри расчетной области, например, в результате испарения или горения (evaporated trajectories).

4. Частицы, расчет траекторий которых становится невозможным в связи с возникновением численных или другого рода ошибок (aborted trajectories).

Частицы, относящиеся к одному из перечисленных типов, исключаются из расчета.

3.3. Число частиц. В случае пренебрежения влиянием дисперсной фазы на течение несущего газа ограничения на число частиц N_p , участвующих в моделировании, определяются, в основном, из статистических соображений. Согласно центральной предельной теореме, статистическая ошибка изменяется как $N_p^{1/2}$. Поскольку каждой частице соответствует одна реализация, то для получения статистически достоверной картины движения примеси требуется достаточно большое число частиц. Например, при $N_p = 10^4$ ошибка составляет примерно 1 %.

При учете влияния частиц на газ число частиц выбирается не произвольным образом, а исходя из концентрации дисперсной фазы или ее массового расхода. Выбранного таким образом числа частиц обычно достаточно, чтобы обеспечить статистическую точность моделирования. С вычислительной точки зрения не всегда удается удовлетворить этому критерию, поскольку требуемое число частиц оказывается за пределами возможностей вычислительной техники.

3.4. Расчет источниковых членов. Расчет источниковых членов остается одной из основных проблем в прямом численном моделировании двухфазных течений [5].

В случае когда влияние частиц на течение газа не учитывается, вычислительная процедура расщепляется на расчет течения газа с последующим расчетом траекторий частицы в известном газодинамическом поле (рис. 3 а). Частицы обычно инжектируются только из тех точек расчетной области, которые представляют интерес для исследования особенностей решаемой задачи.

При учете обратного влияния примеси производится расчет поля течения чистого газа, после чего рассчитываются траектории частиц и вычисляются слагаемые, описывающие межфазный обмен импульсом и энергией. Поле течения газа пересчитывается с учетом слагаемых, описывающих обратное влияние примеси, и снова рассчитываются траектории частиц (в модифицированном поле течения). Процесс повторяется до тех пор, пока не выполняется критерий сходимости (рис. 3 б).

Для сокращения затрат процессорного времени расчет траекторий частиц производится не на каждой итерации, соответствующей расчету поля течения газа, а через их заданное количество.

Источниковые члены в уравнениях Навье–Стокса, учитывающие влияние дисперсной фазы, рассчитываются как произведение счетной концентрации частиц на интенсивности межфазного обмена импульсом, приходящиеся на одну частицу ($F_{pi} = n_{pi} f_{pi}$). Суммарное воздействие частиц на среду определяется суммированием вкладов от всех частиц, находящихся в данной ячейке: $F_p = \sum_{i=1}^M F_{pi}$, где M — число фракций частиц.

Концентрация дисперсной фазы рассчитывается либо из уравнения неразрывности, записанного в

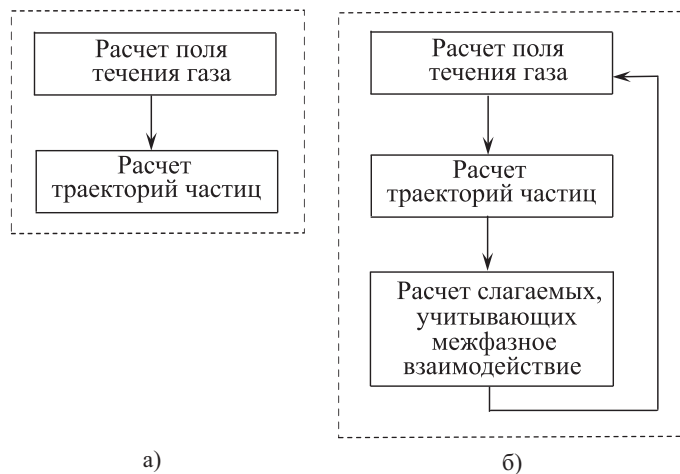


Рис. 3. Реализация вычислительной процедуры без (а) и с учетом влияния частиц на газ (б)

лагранжевых переменных, либо при помощи статистической обработки информации о движении отдельных частиц [3] (посредством осреднения по ячейке расчетной сетки).

Изменение источниковых членов от узла к узлу в некоторых случаях приводит к неустойчивости вычислительной процедуры. Для улучшения сходимости при вычислении слагаемых, описывающих влияние дисперсной фазы на газ, используется нижняя релаксация $\Phi^{n+1} = \Phi^n + \omega (\Phi^* - \Phi^n)$, где ω — релаксационный параметр.

Для устранения неустойчивости при осреднении параметров дисперсной фазы привлекаются соседние узлы и контрольные объемы сетки, а также используется взвешенное среднее по соседним контрольным объемам. Такой подход обеспечивает более гладкое изменение источниковых членов по узлам расчетной сетки.

3.5. Улучшение сходимости. Для улучшения сходимости итерационного процесса увеличивается число частиц, участвующих в моделировании, что способствует более гладкому изменению источниковых членов, описывающих межфазное взаимодействие. Кроме того, уменьшаются показатели релаксации и увеличивается число итераций, производимых для обновления поля течения газа между траекторными расчетами пробных частиц.

В случае нестационарных расчетов на основе неявной схемы подытерации проводятся на каждом шаге интегрирования уравнений движения частицы по времени. При использовании явной схемы частицы продвигаются вперед в конце шага интегрирования по времени.

3.6. Источники ошибок. Ошибки, возникающие при численной реализации лагранжевого подхода, разделяются на следующие типы.

1. Ошибка, связанная с расчетом скорости газа в узлах сетки. Она определяется подходом и конечно-разностными схемами, используемыми для дискретизации уравнений Навье–Стокса, и не зависит от реализации лагранжевого подхода к описанию движения дисперсной фазы.

2. Ошибка, связанная с восполнением (интерполяцией) параметров газа в точках, лежащих на траектории частицы.

3. Ошибка, связанная с интегрированием уравнений, описывающих движение частицы. Исследования показывают, что в регулярных ламинарных течениях ошибка, связанная с определением положения частицы, увеличивается как t^2 , а в хаотических течениях — по экспоненциальному закону.

4. Ошибка, связанная с определением положения частицы (локализацией частицы) при использовании локальной системы координат.

5. Ошибка, возникающая при использовании в расчетах конечного числа частиц. Ошибка имеет статистический характер и уменьшается при увеличении числа частиц.

Суммарный вклад каждого типа ошибок определяет общую погрешность алгоритма.

3.7. Шаг по времени. При интегрировании уравнений, описывающих движение частицы, обычно определяется максимальное число шагов по времени. Расчет траектории частицы прерывается, если частица выходит за пределы расчетной области или достигается заданное число шагов по времени (в этом случае частица остается в пределах расчетной области).

Для управления шагом интегрирования по времени в каждом контрольном объеме задается масштаб длины Δs . Шаг по времени вычисляется исходя из масштаба Δs и скоростей газовой и дисперсной фаз:

$$\Delta t = \frac{\Delta s}{|\mathbf{v} + \mathbf{v}_p|}. \text{ Длина } \Delta s \text{ представляет собой расстояние, которое пройдет частица в данном контрольном объеме до следующего обновления траектории. Малые значения } \Delta s \text{ увеличивают точность расчета траектории частицы и источниковых членов, описывающих межфазное взаимодействие. При этом положение частицы рассчитывается всегда, когда она входит или покидает контрольный объем, независимо от заданного значения масштаба } \Delta s.$$

Вместо масштаба Δs для управления шагом интегрирования по времени задается также значение коэффициента λ , позволяющего рассчитать шаг интегрирования по времени в виде числа шагов, необходимых для пересечения частицей контрольного объема, занятого жидкостью. В этом случае $\Delta t = \frac{\Delta t^*}{\lambda}$, где Δt^* — время пересечения частицей контрольного объема.

Имеются алгоритмы с адаптивным выбором шага интегрирования по времени в зависимости от величины и направления скорости несущего потока, а также метод, основанный на сгущении сетки таким образом, чтобы частица в каждый момент времени оказывалась в следующей по направлению ее движения ячейке сетки [6].

Шаг по времени требует уменьшения при попадании частицы в рециркуляционные зоны, а в случае испаряющихся или горящих частиц — при сильных колебаниях температуры частицы.

4. Локализация частицы. Локализация частицы в данной ячейке или контрольном объеме неструк-

турированной сетки требуется для того, чтобы восполнить параметры газодинамического расчета. Для локализации частицы имеется ряд критериев и алгоритмов, отличающихся друг от друга степенью эффективности и сложностью реализации.

Критерий 1. Рассмотрим точку P на треугольной сетке. Представим вектор однородных координат точки P в виде разложения по базисным функциям $\mathbf{x}_P = \mathbf{X}\mathbf{N}$, где \mathbf{X} — матрица, состоящая из однородных координат узловых точек ячейки сетки, \mathbf{N} — вектор, состоящий из функций формы.

В двумерном случае (сетка состоит из треугольников) имеется три функции формы, а в трехмерном случае (сетка состоит из тетраэдров) — четыре функции формы. Функции формы удовлетворяют условию $\sum_i N_i = 1$. Для нахождения функций формы необходимо решить уравнение $\mathbf{N} = \mathbf{X}^{-1}\mathbf{x}_P$, где

$$\mathbf{x}_P = \begin{pmatrix} x_P \\ y_P \\ z_P \\ 1 \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \\ 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{N} = \begin{pmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \end{pmatrix}.$$

Точка P лежит внутри контрольного объема при условии, что $\min \{N_i, 1 - N_i\} \geq 0$ для $\forall i$. Проводится цикл по всем контрольным объемам, для каждого из которых находятся функции формы. Цикл продолжается до тех пор, пока не выполняется критерий поиска.

В случае сетки с ячейками сложной формы каждый контрольный объем разбивается на треугольники (в двумерном случае) или тетраэдры (в трехмерном случае), для которых критерий локализации известен. Частица находится внутри данного контрольного объема при условии, что она лежит внутри одного из простых элементов, полученных в результате разбиения [7].

Для скошенных сеток и сеток с контрольными объемами, имеющими криволинейные границы, такой подход приводит к ошибкам. К преимуществам подхода относится его сравнительно простая векторизация, обеспечивающая хорошие показатели производительности [4].

Критерий 2. Узлы контрольного объема соединяются с точкой, в которой находится частица. Сумма контрольных объемов, полученных в результате разбиения, сравнивается с исходным контрольным объемом. Частица находится внутри данного контрольного объема при условии, что сумма контрольных объемов, полученных при разбиении, равняется объему исходной ячейки.

Этот подход приводит к ошибкам на растянутых сетках, используемых в расчетах течений в пограничном слое, из-за неточности определения объема ячейки сложной формы [8].

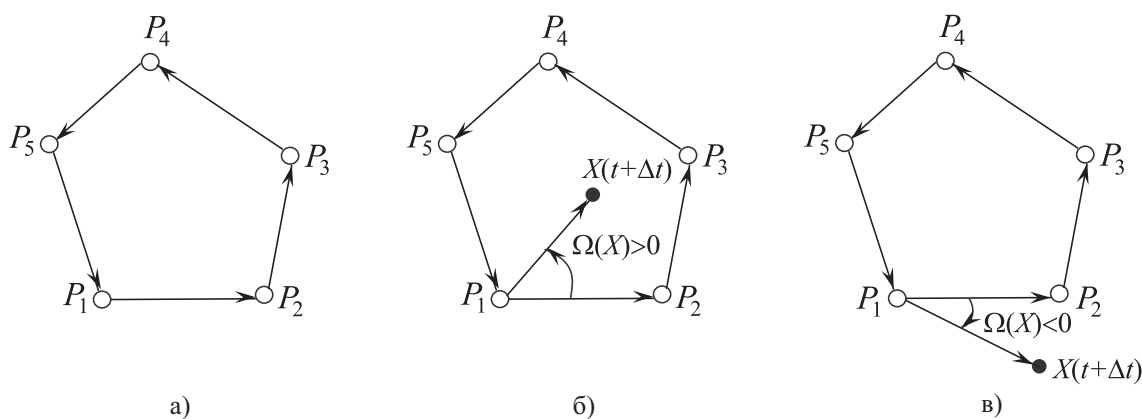


Рис. 4. Критерий P2L в двумерном случае

Критерий 3. Точка, в которой находится частица, проектируется на грани контрольного объема. Полученные вектора сравниваются со всеми внешними нормальными к граням контрольного объема. Частица находится в контрольном объеме при условии, что проекции частицы на его грани совпадают с внешними нормальными.

Данный подход демонстрирует хорошие показатели точности, в том числе и на сетках со скошенными ячейками [8].

Критерий 4. Ячейка двумерной сетки представляется в виде многоугольника, который определяется перечислением его вершин против часовой стрелки [9]: $P = \{P_i = (x_i, y_i) \mid i = 1, 2, \dots, n\}$, где n — число узлов в ячейке.

Для определения того, находится ли частица внутри ячейки, перебираются все грани ячейки против часовой стрелки и проверяется, лежит ли частица слева от всех ее граней (условие particle-to-the-left, или P2L).

Условие P2L проверяется для каждой грани ячейки при помощи вычисления скалярного произведения вектора $P_i P_{i+1}$ (рис. 4 а), задающего направление касательной к грани, и вектора $P_i X(t + \Delta t)$, где $X(t + \Delta t)$ характеризует положение частицы в конце шага интегрирования по времени. Скалярное произведение дает величину

$$\Omega_i = (x_{i+1} - x_i)[y(t + \Delta t) - y_i] - [x(t + \Delta t) - x_i](y_{i+1} - y_i).$$

Значение $\Omega_i > 0$ указывает на то, что частица находится слева от грани (рис. 4 б), а значение $\Omega_i < 0$ — на то, что частица лежит справа от грани (рис. 4 в). При $\Omega_i = 0$ частица лежит на грани.

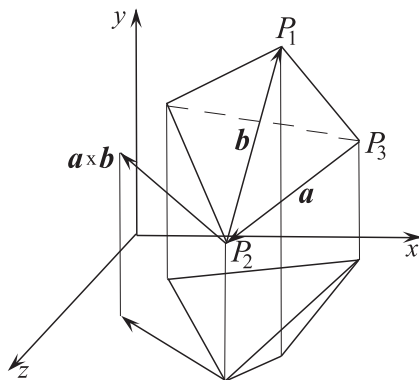


Рис. 5. Критерий P2L в трехмерном случае

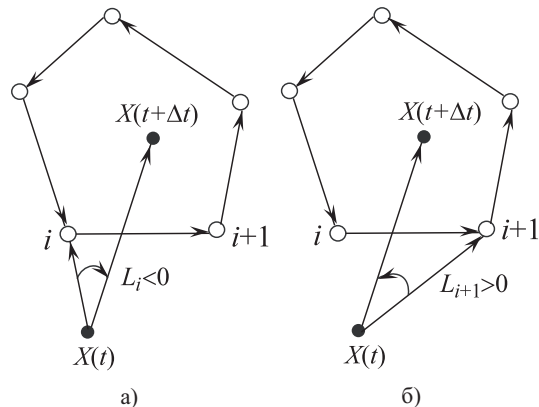


Рис. 6. Критерий T2L

Обобщение критерия P2L на трехмерный случай затрудняется тем, что грань ячейки не всегда является плоской. В трехмерном случае грань ячейки представляется в виде множества узлов, которые обходятся против часовой стрелки, если смотреть на них с внешней стороны ячейки (рис. 5). При таком порядке узлов вектор, полученный в результате векторного произведения векторов $P_{i+2} P_{i+1}$ и $P_{i+1} P_i$, направлен внутрь ячейки. Условие выполнения критерия P2L для каждого узла i ячейки записывается в виде $\text{sign} [(P_{i+2} P_{i+1} \times P_{i+1} P_i) P_{i+1} X(t + \Delta t)] > 0$ [9].

Критерий 5. Проверяется условие пересечения отрезка траектории частицы с гранью ячейки (условие trajectory-to-the-left, T2L). Для этого используется z -компонента векторного произведения векторов $X(t) P_i$ и $X(t) X(t + \Delta t)$ (рис. 6). В узле i имеем

$$L_i = [x_i - x(t)] [y(t + \Delta t) - y(t)] - [x(t + \Delta t) - x(t)] [y_i - y(t)],$$

при этом $L_i > 0$, если траектория частицы лежит слева от данного узла (узла $i + 1$ на рис. 6 а), и $L_i < 0$ — в противном случае (траектория частицы лежит справа от узла i на рис. 6 б). Когда величины L_i и L_{i+1} , рассчитанные в узлах i и $i + 1$, имеют различные знаки, траектория частицы пересекает грань, соединяющую узлы i и $i + 1$.

5. Поиск частицы. При реализации дискретно-траекторного подхода на каждом шаге по времени требуется найти ячейку эйлеровой сетки, в которой находится частица, а также ячейку, следующую по ходу движения частицы.

5.1. Структурированная сетка. Использование обобщенного преобразования координат позволяет провести поиск частицы не в физической плоскости (x, y) с криволинейными границами, а на регулярной сетке в вычислительном пространстве (ξ, η) , ячейки которой представляет собой прямоугольники.

При организации вычислений в пространстве (ξ, η) отпадает необходимость в выявлении следующей по ходу движения частицы ячейки сетки, но приводит к снижению точности интегрирования, поскольку метрические параметры сеточного шаблона не являются аналитическими функциями. Снижение точности имеет место также в областях с искривленными координатными линиями.

Зная положение частицы в физической плоскости (x, y) , легко определить, в какой ячейке криволинейной сетки она находится. При этом возникают трудности с фиксацией момента выпадения частицы на стенку или вылета частицы за пределы расчетной области.

Для устранения перечисленных трудностей интегрирование уравнений движения частицы производится как в физическом, так и в вычислительном пространствах. Поскольку уравнения для координат не связаны с уравнениями для скорости, то интегрирование уравнений для скорости производится в физическом пространстве, а уравнений для координат — в вычислительной плоскости [3].

5.2. Неструктурированная сетка. На неструктурированной сетке требуется разработка специальных вычислительных алгоритмов и критериев. Ряд из них приводится и обсуждается в [10].

5.2.1. Итеративные алгоритмы. В итеративных алгоритмах для поиска частицы выбирается любая ячейка сетки, а положение частицы в момент времени t^{n+1} ищется при помощи последовательных переходов по связанным с ней ячейкам.

5.2.2. Алгоритмы направленного поиска. С вычислительной точки зрения более эффективными являются итеративные алгоритмы с направленным поиском.

Основные типы. Поскольку положения частицы в моменты времени t^n и t^{n+1} не являются статистически независимыми, то в качестве начальной ячейки для поиска выбирается ячейка, найденная ранее (для положения частицы в предыдущий момент времени). На практике обычно используется ряд способов поиска ячейки по положению частицы внутри контрольного объема и некоторой начальной ячейке (рис. 7).

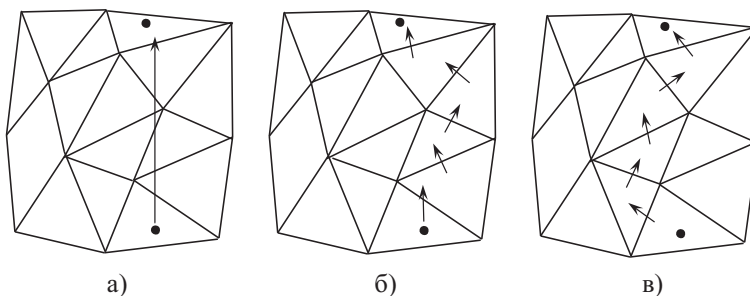


Рис. 7. Варианты локализации ячейки расчетной сетки

1. Проводится прямая через положение частицы в моменты времени t^n и t^{n+1} (рис. 7 а). Поиск осуществляется вдоль этой прямой к цели, при этом необходимо корректно обрабатывать ситуации, когда на пути встречаются узлы и коллинеарные ребра.

2. Движение осуществляется по одному шагу, причем на каждом временном шаге проводится прямая через центр текущей ячейки и положение частицы в момент времени t^{n+1} (рис. 7 б). Переход к соседней ячейке осуществляется через ребро, которое пересекает построенная прямая.

3. Движение осуществляется по одному шагу. На каждом шаге переход к соседней ячейке производится через такое ребро, что целевая точка и вершина текущей ячейки, противоположная выбираемому пересекаемому ребру, лежат по разные стороны от прямой, определяемой данным ребром (рис. 7 в).

Конкретные реализации рассматриваемых ниже алгоритмов различаются используемым критерием локализации (или их комбинацией), обработкой исключительных ситуаций и рядом других деталей.

Алгоритм 1. В наиболее простом случае производится последовательный перебор всех контрольных объемов сетки (при переборе используется матрица связности), к каждому из которых применяется тот или иной критерий локализации частицы. Несмотря на простоту, подход является достаточно дорогостоящим для сеток большой размерности и большого числа частиц. В худшем случае приходится перебирать все ячейки, поэтому трудоемкость поиска составляет $O(N)$.

Алгоритм 2. Выбирается узел сетки, ближайший к точке, в которой находится частица, и рассматривается контрольный объем, окружающий эту точку. При неудачной попытке (критерий локализации не выполняется) перебираются контрольные объемы, окружающие соседние узлы сетки.

Алгоритм 3. Более эффективный подход к поиску частицы состоит в использовании координат частицы в начальный и несколько последующих моментов времени (алгоритм neighbor-to-neighbor). Положение частицы после нескольких начальных шагов по времени дает хорошее приближение для последующего поиска, который проводится в направлении движения частицы (рис. 8). В случае неудачи, когда частица

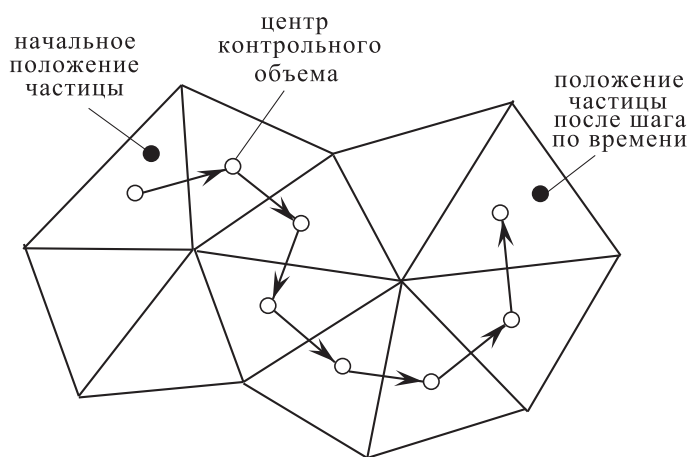


Рис. 8. Поиск частицы на неструктурированной сетке (алгоритм 3)

не находится за 10–15 попыток (например, при круговой траектории поиска), применяется обычный алгоритм перебора. Для определения того, находится ли частица внутри контрольного объема, используется один из критериев локализации.

Производительность алгоритма в существенной степени зависит от удачного выбора начального приближения [4, 7].

В ряде случаев алгоритм дает круговую траекторию поиска. Такая ситуация имеет место, например, при использовании осесимметричной структурированной сетки, имеющей особенность на линии симметрии (расчет течения в круглом канале). Для предотвращения заикливания алгоритма разумно модифицировать сетку около линии симметрии.

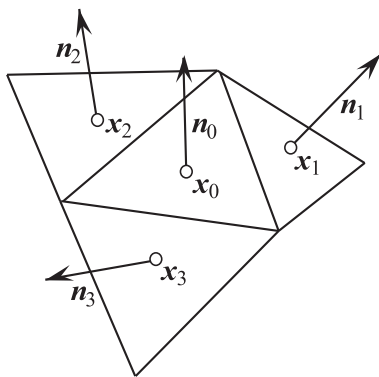


Рис. 9. Поиск частицы в приграничном контрольном объеме (алгоритм 3)

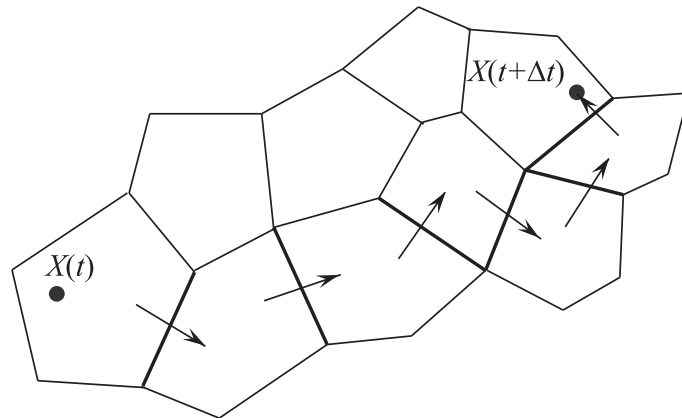


Рис. 10. Поиск частицы на неструктурированной сетке (алгоритм 4)

Алгоритм приводит также к проблемам в приграничных контрольных объемах, имеющих криволинейную вогнутую форму с большим числом точек на границе области [7]. В этом случае число попыток оценивается как $O(N_b^2)$, где N_b — число граничных узлов. Для вогнутой поверхности определяется расстояние по нормали между узлом, лежащим на границе области, и геометрическими центрами приграничных контрольных объемов $d = \alpha |\min \{0, \mathbf{n} \cdot (\mathbf{x}_0 - \mathbf{x}_i)\}|$, где \mathbf{n} — внешняя нормаль и $0.5 < \alpha < 1.5$ (рис. 9). Частица находится внутри контрольного объема, если расстояние от частицы до границы области не превосходит d .

Алгоритм 4. В данном алгоритме для локализации частицы используется критерий P2L (критерий 4), а поиск частицы производится в направлении ее движения (рис. 10).

Величина Ω_i рассчитывается для всех граней ячейки до тех пор, пока одна из граней не даст отрицательного значения (на рис. 10 эти грани указываются утолщенными линиями). В случае $\Omega_i < 0$ осуществляется переход к ячейкам, которые включают грань с отрицательным значением Ω_i . Перебор граней заканчивается, когда $\Omega_i > 0$ для всех граней данной ячейки (частица находится внутри ячейки).

В трехмерном случае недостатки алгоритма, связанные с теньвым эффектом (заикливание алгоритма в случае круговой траектории поиска), усиливаются из-за неточностей, вносимых представлением каждой грани в виде плоскости [9].

Алгоритм 5. Поиск ячейки, в которой находится частица, проводится путем перебора всех ячеек, которые пересекает частица за данный интервал времени [11] (рис. 11). Шагу интегрирования по времени Δt соответствует участок траектории $X(t)X(t + \Delta t)$. Управляющим критерием служит выбор грани ячейки, которую пересекает частица при переходе в соседнюю ячейку.

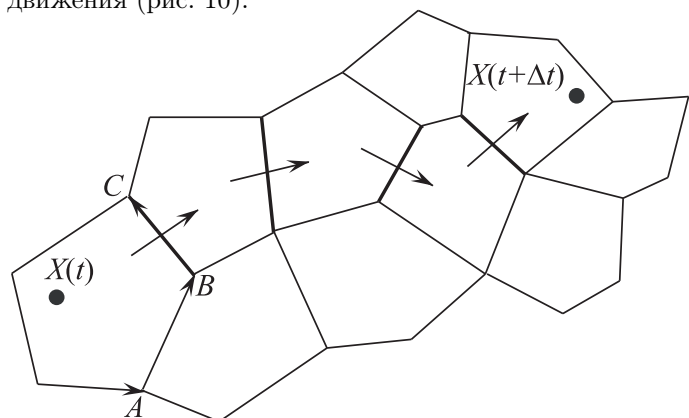


Рис. 11. Поиск частицы на неструктурированной сетке (алгоритм 5)

Сначала с использованием критерия 4 проверяется, находится ли частица слева от каждой грани данной ячейки [9]. При выполнении критерия для каждой грани поиск частицы прекращается. В противном случае ищется грань, которую пересекает частица при переходе в следующую ячейку. Для этого выбираются грани ячейки, справа от которых находится частица (грани AB и BC на рис. 11). После этого рассчитываются точки пересечения траектории частицы $X(t)X(t + \Delta t)$ с этими гранями. Из этих граней выбирается та грань, в пределах которой лежит найденная точка пересечения (грань BC на рис. 11). Потом производится переход к ячейке, содержащей грань, которую пересекает частица. Процесс повторяется до тех пор, пока не выполняется критерий поиска.

Наиболее затратный шаг алгоритма состоит в нахождении грани, через которую частица покидает данную ячейку [11] (это требует расчета координат точек пересечения нескольких линий).

Алгоритм 6. Для определения грани, через которую частица выходит из ячейки, используется критерий 4 [9]. Вместо нахождения точки пересечения частицы с гранью, как в алгоритме 5, проверяется условие T2L (критерий 5).

Данный алгоритм реализуется в виде следующей последовательности шагов [11, 12] (рис. 12).

Проверяется, пересекает ли траектория частицы $X(t)X(t + \Delta t)$ одну из граней данной ячейки. Для этого для двух соседних узлов проверяется условие T2L. Для ячейки 1 на рис. 12 частица пересекает грани BC и FA .

В случае если частица пересекает грань ячейки, проверяется условие P2L для данной грани. Если частица находится справа от грани (критерий P2L дает отрицательное значение), то грань, через которую частица покидает данную ячейку, найдена (для ячейки 1 это грань BC на рис. 12). Производится выход из цикла и переход к соседней ячейке, которая содержит найденную грань. Для новой ячейки производится такая же последовательность шагов. Завершение цикла по всем граням и выполнение условия P2L (критерий дает положительное значение) для всех граней, которые пересекает частица (для ячейки 2 на рис. 12 это грань IJ и грань BC), означает, что ячейка, в которой находится частица в конце шага интегрирования по времени, найдена.

Критерий пересечения траектории частицы с гранью не выполняется, если частица пересекает грань в одном из ее узлов (критерий 5 дает $L_i = 0$ в узле i или $L_{i+1} = 0$ в узле $i + 1$). Для устранения такой ситуации предполагается, что частица пересекает грань, соединяющую узлы i и $i + 1$, если $L_i L_{i+1} \leq 0$. В трехмерном случае вместо нахождения точки пересечения траектории частицы с гранью произвольной формы [11] производится расчет ряда простых алгебраических соотношений [12].

Для того чтобы проверить, пересекает ли частица грань ячейки, проверяется относительная ориентация ребер грани и траектории частицы (ребро P_1P_2 нижней грани на рис. 13). Относительная ориентация грани P_1P_2 и траектории частицы определяется плоскостью, содержащей точки P_1 , P_2 и $X(t + \Delta t)$. При этом возможны три ситуации взаимного положения частицы и грани P_1P_2 в момент времени $t + \Delta t$, отмеченные на рис. 13 индексами a (траектория частицы пересекает грань ячейки), b (траектория частицы пересекает ребро) и c (траектория частицы не пересекает грани ячейки).

Траектория частицы пересекает грань данной ячейки, если условие, отмеченное на рис. 13 индексом a ,

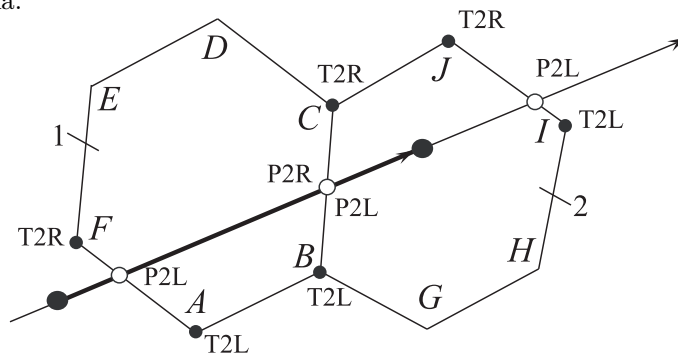


Рис. 12. Поиск частицы на неструктурированной сетке (алгоритм 6)

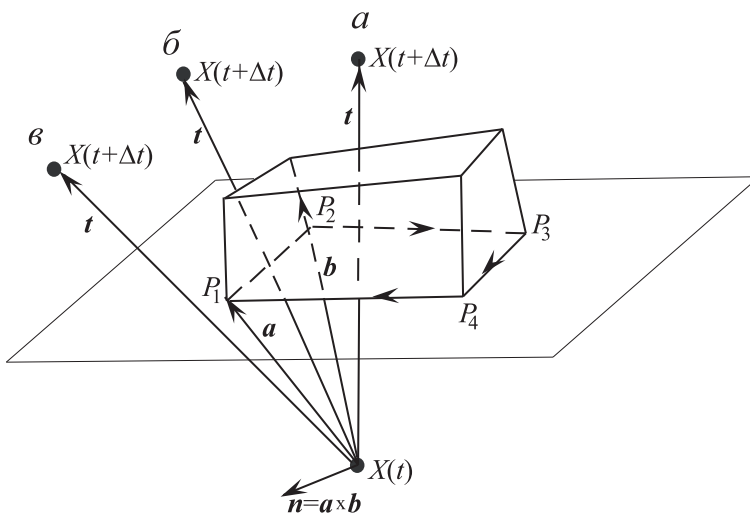


Рис. 13. Относительная ориентация грани и траектории частицы (алгоритм 6)

выполняется для всех ребер $P_i P_{i+1}$ данной грани. Определим вектор $\mathbf{a} = X(t)P_1$ и вектор $\mathbf{b} = X(t)P_2$, а также вектор $\mathbf{n} = \mathbf{a} \times \mathbf{b}$, представляющий собой вектор внешней нормали к плоскости $P_1 P_2 X(t)$ (нормаль является внешней в силу принятого порядка обхода узлов грани). Для каждого возможного положения частицы в момент времени $t + \Delta t$ определим вектор $\mathbf{t} = X(t)X(t + \Delta t)$, задающий направление движения частицы на интервале времени Δt .

Для верхней грани ячейки условие пересечения траектории частицы с гранью выполняется, если $\mathbf{n} \cdot \mathbf{t} > 0$ для всех ребер $P_i P_{i+1}$ (учитывается другой порядок обхода для узлов верхней грани). В зависимости от ориентации векторов \mathbf{n} и \mathbf{t} возможны различные варианты реализации. При $\mathbf{n} \cdot \mathbf{t} < 0$ реализуется вариант, помеченный индексом a , а при $\mathbf{n} \cdot \mathbf{t} = 0$ и $\mathbf{n} \cdot \mathbf{t} > 0$ — варианты, отмеченные индексами b и c .

В трехмерном случае критерий пересечения траекторией частицы грани ячейки определяется знаком скалярного произведения: $\text{sign} \left\{ [X(t)P_i \times X(t)P_{i+1}] \cdot \mathbf{t} \right\} = \text{const}$ (знак выражения не изменяется при переходе от одного узла к другому). Знак скалярного произведения $\mathbf{n} \cdot \mathbf{t}$ дает дополнительную информацию. При нумерации узлов всех граней данной ячейки против часовой стрелки $\text{sign}(\mathbf{n} \cdot \mathbf{t})$ говорит о том, покидает ли частица ячейку или входит в нее через данную грань. При $\mathbf{n} \cdot \mathbf{t} > 0$ частица покидает ячейку, а при $\mathbf{n} \cdot \mathbf{t} < 0$ частица входит в ячейку. После проверки, пересекает ли частица грани данной ячейки, необходимо проверить, находится ли частица в конце шага интегрирования по времени внутри ячейки или нет. Для этого используется один из критериев локализации.

Алгоритм 7. На неструктурированной сетке используется локальная система координат, связанная с ячейкой, что позволяет обеспечить эффективность вычислительной процедуры. В то же время, использование локальной системы координат требует расчета координат точки выхода частицы из ячейки.

Координаты частицы в локальной системе координат находятся из решения уравнений

$$\frac{d\xi}{dt} = J^{-1}V_x, \quad \frac{d\eta}{dt} = J^{-1}V_y.$$

Здесь J — якобиан перехода от локальной (ξ, η) к глобальной (x, y) системе координат.

Для определения точки выхода частицы из ячейки находит также применение полиномиальное приближение траектории частицы с последующим расчетом точки пересечения с гранью [6, 13]. Однако необходимость вычисления коэффициентов полинома высокой степени приводит к дополнительным затратам времени.

Для простоты рассмотрим сетку, состоящую из тетраэдров. Предположим, что в момент времени t_0 частица имеет координаты $\mathbf{x}_0 = \{x_0, y_0, z_0\}$, скорость $\mathbf{v}_0 = \{u_0, v_0, w_0\}$ и находится в ячейке $E_1 = \{P_1, P_2, P_3, P_4\}$, которая задается перечислением узловых точек. Положение узловой точки P_i характеризуется вектором $\mathbf{x}_i = \{x_i, y_i, z_i\}$.

Координаты частицы $\mathbf{x} = \{x, y, z\}$ в момент времени t_1 находятся из соотношения $\mathbf{x} = \mathbf{x}_0 + \mathbf{v}_0 \Delta t$, где $\Delta = t_1 - t_0$. Траектория частицы на интервале времени Δt представляется в виде прямой линии (рис. 14).

Рассмотрим случай, когда частица с координатами $\mathbf{x} = \{x, y, z\}$ находится внутри элемента E_1 . Выражая координаты частицы через локальные координаты $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$, где $0 \leq \lambda_i \leq 1$ для $\forall i$, получим

$$\begin{aligned} x &= x_0 + v_x \Delta t = \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 + \lambda_4 x_4, \\ y &= y_0 + v_y \Delta t = \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3 + \lambda_4 y_4, \\ z &= z_0 + v_z \Delta t = \lambda_1 z_1 + \lambda_2 z_2 + \lambda_3 z_3 + \lambda_4 z_4, \end{aligned}$$

где $\sum_{i=1}^4 \lambda_i = 1$. Положим $\lambda_1 = 1 - \lambda_2 - \lambda_3 - \lambda_4$. Тогда величины λ_2, λ_3 и λ_4 находятся из решения системы из трех линейных уравнений. Рассчитанные значения, если они удовлетворяют наложенным ограничениям,

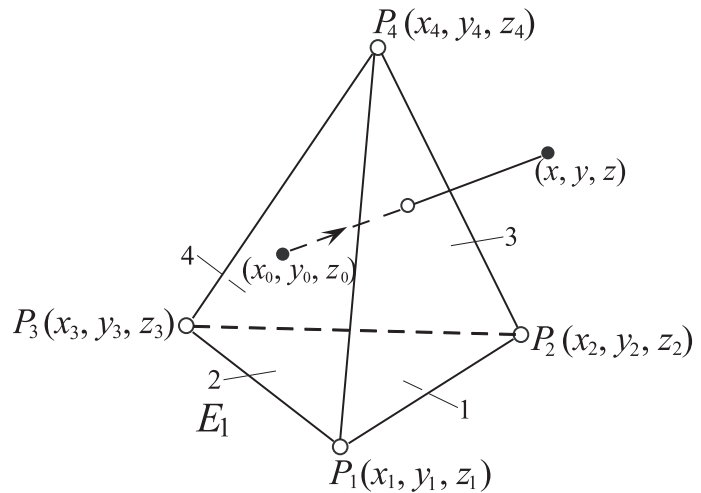


Рис. 14. Положение частицы в моменты времени t_0 и t_1 (частица пересекает грань)

дают положение частицы в новый момент времени и используются для расчета скорости жидкости в

точке, в которой находится частица: $\mathbf{v} = \sum_{i=1}^4 \lambda_i \mathbf{V}_i$, где \mathbf{V}_i — скорости жидкости в узлах ячейки.

Рассмотрим случай, когда частица пересекает грань, определяемую тремя узловыми точками $\{P_{i1}, P_{i2}, P_{i3}\}$. Последовательность $\{i1, i2, i3\}$ определяет ориентацию грани контрольного объема и представляет собой одну из четырех возможных комбинаций $\{1, 3, 2\}$, $\{1, 2, 4\}$, $\{2, 3, 4\}$ или $\{1, 4, 3\}$ (рис. 14), при этом выполняется условие

$$\begin{aligned} x_c &= x_0 + v_x \delta t = \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3; \\ y_c &= y_0 + v_y \delta t = \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3; \\ z_c &= z_0 + v_z \delta t = \lambda_1 z_1 + \lambda_2 z_2 + \lambda_3 z_3. \end{aligned}$$

Точка P_i имеет координаты $\{x_i, y_i, z_i\}$. Весовые коэффициенты удовлетворяют условию $\sum_{i=1}^3 \lambda_i = 1$. В случае пересечения частицей грани $\{P_{i1}, P_{i2}, P_{i3}\}$, когда координаты $\{x, y, z\}$ лежат на грани ячейки, выполняется условие $0 \leq \lambda_i \leq 1$ для $\forall i$ и $\delta t \geq 0$. Весовые коэффициенты представляют собой локальные координаты на грани ячейки.

Положим $\lambda_1 = 1 - \lambda_2 - \lambda_3$. Тогда нахождение величин δt , λ_2 и λ_3 сводится к решению системы из трех линейных уравнений. В случае, если в течение шага интегрирования частица пересекает одну из граней ячейки, то $\delta t \leq \Delta t$.

Единственность решения не гарантируется тогда, когда начальная точка $\{x_0, y_0, z_0\}$ и конечная точка $\{x, y, z\}$ лежат на одной грани и $\delta t = 0$ (рис. 15). В этом случае выбирается решение, удовлетворяющее условию $\delta t > 0$, а точка пересечения частицей грани находится с использованием значений $\{\lambda_1, \lambda_2, \lambda_3\}$ из решения системы уравнений

$$\begin{aligned} x_c &= \lambda_1 x_{i1} + \lambda_2 x_{i2} + \lambda_3 x_{i3}; \\ y_c &= \lambda_1 y_{i1} + \lambda_2 y_{i2} + \lambda_3 y_{i3}; \\ z_c &= \lambda_1 z_{i1} + \lambda_2 z_{i2} + \lambda_3 z_{i3}. \end{aligned}$$

Здесь $\{x_{i1}, x_{i2}, x_{i3}\}$ — координаты узловых точек грани, которую пересекает частица.

При пересечении частицей одной из граней контрольного объема величина δt не превосходит шага интегрирования по времени Δt , и для последующего расчета используется разность $\Delta t - \delta t$. В качестве начальной точки для последующего расчета берется точка $\{x_c, y_c, z_c\}$.

Далее ищется ячейка, в которой оказывается частица (это ячейка, имеющая общую грань с предыдущей ячейкой). При программной реализации для этого используется матрица связности $E(n_e, n_s)$, где n_e — число ячеек и n_s — число граней ($n_s = 4$ для тетраэдра). С использованием матрицы связности следующий контрольный объем сетки i_2 , имеющий с предыдущим контрольным объемом i_1 общую грань i_s , находится как $i_2 = E(i_1, i_s)$. Матрица связности создается один раз перед инъекцией частиц. Значение $E(i_e, i_s) = 0$ означает, что данная грань лежит на границе расчетной области. При пересечении частицей такой грани используется условие отражения частицы от стенки $\hat{\mathbf{v}}_p = \mathbf{v}_p - 2(\mathbf{v}_p \cdot \mathbf{n})\mathbf{n}$ или условие прилипания $\hat{\mathbf{v}}_p = 0$ (здесь символ “крышка” соответствует частице, отраженной от стенки).

Необходимо также учесть особый случай, соответствующий пересечению частицей ребра ячейки (рис. 16), когда скорость частицы направлена внутрь грани (internal reflection). В этом случае условие $\delta t > 0$ для данной грани не выполняется.

Другая исключительная ситуация состоит в получении малых отрицательных значений ($\lambda_i \sim -10^{-10}$) из-за вычислительных ошибок. Такие значения λ_i не удовлетворяют наложенным ограничениям в обоих рассмотренных случаях.

Вычислительная процедура реализуется в виде следующей последовательности шагов.

1. Задается начальное положение \mathbf{x}^n и скорость частицы \mathbf{v}^n на грани или внутри ячейки в момент времени t^n .

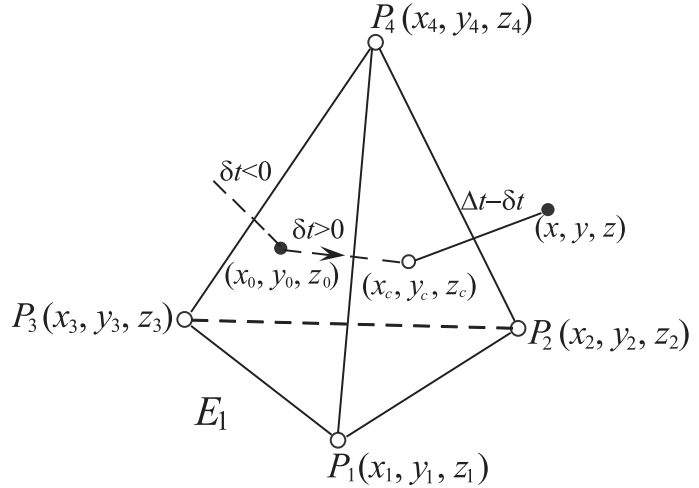


Рис. 15. Положение частицы в моменты времени t_0 и t_1 (отсутствие единственности решения)

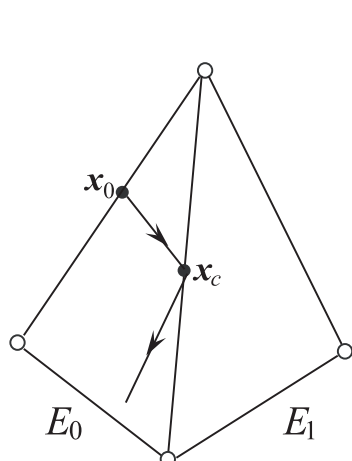


Рис. 16. Отражение частицы от ребра грани

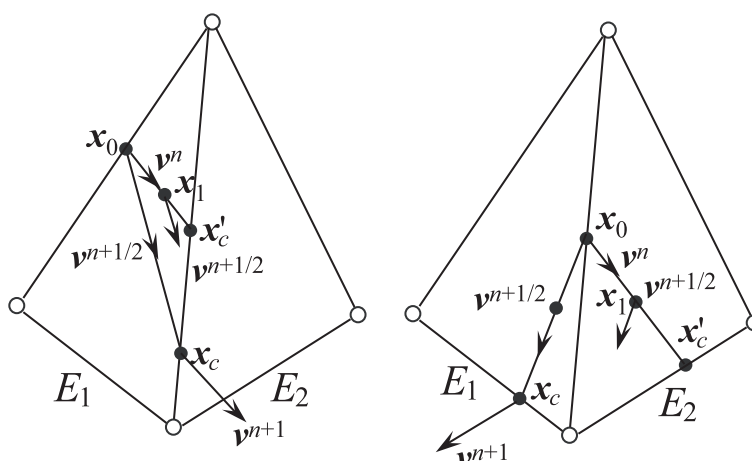


Рис. 17. Повышение точности алгоритма

2. Рассчитывается шаг интегрирования по времени $\Delta t = t^{n+1} - t^n$ и проверяется условие окончания расчета $t > t_f$ или $n > n_f$, где t_f и n_f задают интервал времени, на котором производится моделирование, и максимальное число шагов по времени. Если условие прерывания выполняется, то производится переход к шагу 7.

3. Рассчитывается новое положение частицы путем интегрирования уравнений движения (используется одна из схем численного интегрирования).

4. Рассчитываются значения λ_i для $i = 1, 2, 3, 4$ и проверяется, находится ли частица внутри ячейки. При выполнении критерия локализации текущий шаг по времени завершается и находится скорость жидкости в точке, лежащей на траектории частицы, в момент времени t^{n+1} при помощи соответствующей интерполяционной формулы. Затем рассчитывается скорость частицы в новый момент времени и осуществляется переход к шагу 2. Если условие не выполняется, то частица пересекает одну из граней ячейки.

5. Для каждой грани ячейки $i_s = 1, 2, 3, 4$ находятся величины λ_i для $i = 1, 2, 3$ и шаг по времени δt , и проверяются условия, наложенные на λ_i и δt . Проверяется возникновение исключительных ситуаций. В случае отражения частицы от ребра грани (внутреннее отражение) осуществляется переход к шагу 4 (возврат к предыдущей ячейке).

6. Рассчитываются координаты точки пересечения частицы с гранью x_c и модифицируется шаг по времени. Производится переход к следующей ячейке, рассчитывается скорость газа в точке, в которой располагается частица, и находится скорость частицы. Осуществляется переход к шагу 3.

7. Окончание счета.

Увеличение точности алгоритма до второго порядка достигается за счет дробления шага интегрирования по времени (рис. 17). Дополнительная точка добавляется в момент времени $t^{n+1/2}$ посередине прямолинейного отрезка, представляющего траекторию частицы (рис. 17 слева, частица находится внутри одной и той же ячейки). Рассчитывается предварительная точка x'_c , а затем точка x_c . В случае если предварительная точка x'_c находится в другой ячейке, то требуется дополнительная проверка (рис. 17 справа, частица находится внутри другой ячейки). Для интерполяции скорости газа берется точка x_i .

5.2.3. Алгоритмы индексированного поиска. Все ячейки треугольной сетки заносятся в некоторую древовидную структуру, с помощью которой осуществляется поиск частицы [10].

Алгоритм 1. Для всех ячеек вычисляется минимальный объемлющий прямоугольник (в трехмерном случае — параллелепипед) со сторонами, параллельными осям координат, и заносится в R-дерево.

Каждая вершина R-дерева имеет переменное количество элементов. Каждый элемент нелистовой вершины хранит два поля данных: способ идентификации дочерней вершины и ограничивающий прямоугольник, охватывающий все элементы этой дочерней вершины.

Каждый элемент листовой вершины хранит два поля данных: способ идентификации данных (либо сами эти данные), описывающих объект, и ограничивающий прямоугольник этого объекта.

Алгоритмы вставки и удаления используют эти ограничивающие прямоугольники для того, чтобы обеспечить размещение близкорасположенных объектов в одну листовую вершину. В частности, новый объект попадает в ту листовую вершину, для которой требуется наименьшее расширение ее ограничива-

ющего прямоугольника.

Для поиска ячейки, в которую попадает частица, выполняется стандартный точечный запрос к R-дереву; в результате получается список ячеек, объемлющие прямоугольники которых находятся в данной точке. Из них выбирается та ячейка, внутрь которой попадает частица. Трудоемкость поиска треугольника в R-дереве в худшем случае составляет $O(N)$, а в среднем — $O(\log N)$.

Алгоритм 2. Центры треугольников заносятся в kD-дерево, представляющее собой обобщение бинарного дерева, но ориентированное на многомерные данные. На каждом шаге производится разбиение только вдоль одной плоскости. В качестве плоскости разбиения обычно выбирается плоскость, перпендикулярная оси, вдоль которой ячейка имеет наибольший размер.

Для выполнения поиска ячейки, в которую попадает частица, выполняется нестандартный точечный запрос к kD-дереву. Поиск в дереве производится, начиная с корня и спускаясь вниз до листьев. В случае если потомки текущего узла kD-дерева не покрывают текущую точку, для дальнейшего спуска по дереву выбирается потомок, ближайший к точке поиска. В результате, находится некоторый треугольник, центр которого близок к частице. Трудоемкость поиска частицы в kD-дереве в худшем случае составляет $O(N)$, а в среднем — $O(\log N)$. При необходимости, например в случае круговой траектории поиска, используется простой алгоритм перебора треугольников.

Алгоритм 3. Центры треугольников помещаются в квадротомическое дерево (Quad tree, или Q-tree), которое является более простым в реализации по сравнению с kD-деревом.

Производится декомпозиция пространства на квадратные блоки, каждый из которых делится рекурсивно на четыре вложенных до достижения некоторого уровня (Morton order), обеспечивающего требуемую детальность описания объекта. Для индексирования элементов древовидной структуры используется матрица Мортон (Morton matrix), основанная на кривых и числах Пеано (Peano curves and keys).

Алгоритм 4. Для быстрой локализации точки и поиска треугольника, близкого к исходному, строится специальная структура данных (кэш), основная идея которой заключается в построении некоторого более простого разбиения области, чем триангуляция (обычно используется регулярная сеть квадратов). Процедура поиска сводится к локализации элемента простого разбиения, переходу по ссылке к треугольнику и последующей локализации искомого треугольника.

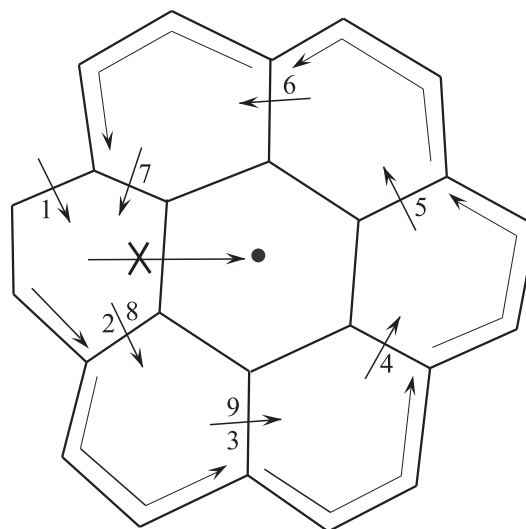


Рис. 18. Круговая траектория поиска (алгоритм 4)

5.3. Достоинства и недостатки некоторых алгоритмов. Несмотря на простоту реализации и вычислительную эффективность, алгоритм 4, разработанный в [9], обладает рядом недостатков. В ряде случаев, один из которых поясняет рис. 18, этот алгоритм не позволяет найти ячейку, в которой находится частица, и приводит к заикливанию, поскольку траектория поиска является круговой (shadowing effect). Обобщение алгоритма 4 на трехмерный случай является достаточно сложным. Алгоритм перебирает большее число ячеек по сравнению с другими алгоритмами, например с алгоритмом 5 [11].

Алгоритм 5, разработанный в [11], позволяет найти частицу за меньшее число шагов. К его недостаткам также относится сложность реализации в трехмерном случае (необходимо определить точку пересечения траектории частицы с гранью ячейки) и достаточно большое общее время, необходимое для отыскания частицы. В случае сложной геометрии нахождение точки пересечения траектории частицы с гранью ячейки произвольной формы представляет собой трудную задачу.

Общие затраты времени на нахождение частицы при использовании алгоритма 5 оказываются примерно в 2–4 раза выше, чем в алгоритме 4, несмотря на то что алгоритм 4 требует перебора несколько большего числа ячеек (примерно на 2%).

Алгоритм 6, разработанный в [12], обладает преимуществами алгоритмов 4 и 5 и устраняет их недостатки. Кроме того, этот алгоритм сравнительно просто обобщается на трехмерный случай, при этом время поиска частицы оказывается пропорциональным числу ячеек, пересекаемых частицей.

Алгоритмы индексированного поиска на практике используются сравнительно редко, что объясняется необходимостью построения и хранения соответствующих древовидных структур данных.

6. Восполнение параметров газа. Корректность решения задачи зависит от точности задания

параметров газа, входящих в уравнение движения пробной частицы. Для построения гладких восполнений газодинамических параметров в точках, лежащих на траектории частицы, по их значениям в узлах эйлеровой сетки используются различные подходы.

6.1. Линейная интерполяция. Линейная интерполяция представляет собой наиболее простой подход к расчету скорости газа в точке, в которой находится частица, и находит широкое применение для структурированных сеток с прямоугольными ячейками.

Скорость, направленная по нормали к грани ячейки, изменяется по линейному закону вдоль направления движения частицы, а тангенциальная скорость остается постоянной. Такой подход приводит к тому, что $v_x = v_x(x)$ и $v_y = v_y(y)$. Он является совместимым с конечно-разностной дискретизацией уравнений Навье–Стокса в узлах структурированной сетки, поскольку при этом выполняется условие сохранения массы.

Кусочно-линейное представление скорости несущего потока позволяет использовать схемы полуаналитического интегрирования уравнений движения пробной частицы. Недостаток подхода связан с тем, что тангенциальная скорость терпит разрыв на гранях ячеек. Другим недостатком линейной интерполяции является ее невысокая точность. Практика численных расчетов показывает, что основные погрешности связаны не с низким порядком интерполяции, а со скошенностью ячеек расчетной сетки, в связи с чем восполнение параметров газа имеет смысл производить в вычислительном пространстве (в расчетах на криволинейных сетках).

6.2. Билинейная интерполяция. Задача интерполяции сводится к построению в каждой ячейке расчетной сетки системы двух линейных уравнений, решение которой получается в конечном виде.

В случае билинейной интерполяции имеем $v_x = v_x(x, y)$ и $v_y = v_y(x, y)$, что не гарантирует условия сохранения массы внутри контрольного объема, но обеспечивает непрерывность поля скорости.

Результаты расчетов показывают, что, несмотря на невыполнение условия сохранения массы, билинейная интерполяция дает более точные результаты, чем линейная интерполяция [6].

6.3. Методы высокого порядка. При отсутствии в вычислительной процедуре механизма адаптации сетки к решению использование гладкой интерполяции, например полиномов третьего или четвертого порядка, приводит к нежелательным осцилляциям и потере физической корректности решения. Использование методов интерполяции высокого порядка требует достаточно точного расчета поля скорости газа.

6.4. Сплайн-интерполяция. Применение кубической сплайн-интерполяции позволяет, помимо вычисления скорости, без дополнительных затрат вычислить значения производных скорости (при использовании линейной интерполяции для этого требуется специальная процедура), которые необходимы для расчета компонент матрицы Якоби.

6.5. Неструктурированная сетка. Для восполнения параметров газа на неструктурированной сетке используются значения, рассчитанные в центрах контрольных объемов [6]. В расчетах на сетках со скошенными ячейками, например в пограничном слое, требуются более точные алгоритмы. Кроме того, контрольные объемы неструктурированной сетки обычно имеют различное число соседей в зависимости от формы элемента (тетраэдр, призма, шестигранник), что усложняет реализацию подхода.

6.6. Метод взвешенных площадей. Формулы, обеспечивающие непрерывность интерполируемой величины при переходе частицы из одной ячейки сетки в другую, строятся также на основе метода взвешенных площадей: $x = \frac{\sum_{i=1}^4 A_i x_i}{\sum_{i=1}^4 A_i}$, где A_i — площадь соответствующего прямоугольника (рис. 19).

шенных площадей: $x = \frac{\sum_{i=1}^4 A_i x_i}{\sum_{i=1}^4 A_i}$, где A_i — площадь соответствующего прямоугольника (рис. 19).

7. Расстояние до стенки. В некоторых моделях, например в случае учета поправки к коэффициенту сопротивления на близость стенки, требуется рассчитать расстояние от частицы до стенки. Расстояние от точки P , в которой находится частица, до точки Q на стенке (рис. 20) оценивается по формуле $y = \mathbf{PA} \cdot \mathbf{n}$, где $\mathbf{n} = \frac{\mathbf{AB} \times \mathbf{AC}}{|\mathbf{AB}| |\mathbf{AC}|}$.

8. Параллелизация. В случае, когда влияние частиц на течение газа и столкновения между частицами не учитываются, траектории частиц рассчитываются независимо друг от друга. Число траекторий частиц, рассчитываемых в данный момент времени, равняется числу процессоров системы. Параллелизация лагранжевого подхода к описанию движения примеси требует, чтобы в траекторных расчетах пробных частиц использовалась информация, содержащаяся в блоке сетки, который относится к данному процессору.

9. Заключение. Рассмотрена реализация ряда вспомогательных алгоритмов, связанных с локализа-

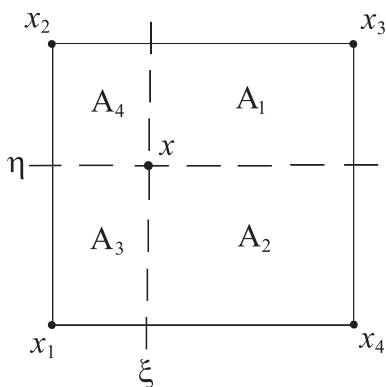


Рис. 19. Метод взвешенных площадей

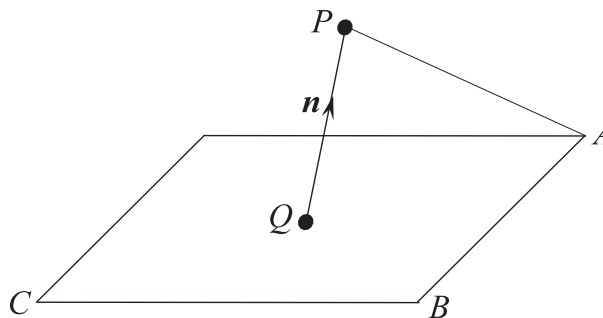


Рис. 20. Расстояние от частицы до стенки

цией частицы в ячейке неструктурированной сетки и поиском контрольного объема, в котором находится частица в данный момент времени. Ряд из приведенных алгоритмов находит применение в итеративных алгоритмах построения триангуляции Делоне, одним из этапов которых является добавление точки в построенную структуру триангуляции и локализация точки (поиск построенного ранее треугольника, в который попадает точка). Сложность итеративных алгоритмов триангуляции во многом определяется трудоемкостью поиска треугольника, в который добавляется новая точка.

СПИСОК ЛИТЕРАТУРЫ

1. Волков К.Н. Стохастическое моделирование движения и рассеивания примеси в механике турбулентных газодисперсных течений // Инженерно-физический журнал. 2004. **77**, № 5. 10–20.
2. Волков К.Н. Стохастические модели движения частицы в турбулентном потоке и их применение для расчета внутренних течений // Инженерно-физический журнал. 2007. **80**, № 3. 136–147.
3. Волков К.Н. Разностные схемы интегрирования пробной частицы в потоке жидкости или газа // Вычислительные методы и программирование. 2004. **5**, № 1. 5–21.
4. Petera J., Weatherley L.R., Hume A.P., Gawrysiak T. A finite element algorithm for particle/droplet trajectory tracking, tested in a liquid-liquid system in the presence of an external electric field // Computers and Chemical Engineering. 2007. **31**, N 11. 1369–1388.
5. Mashayek F., Pandya R.V.R. Analytical description of particle/droplet-laden turbulent flows // Progress in Energy and Combustion Science. 2003. **29**, N 4. 329–378.
6. Pokrajac D., Lazic R. An efficient algorithm for high accuracy particle tracking in finite elements // Advances in Water Resources. 2002. **25**, N 4. 353–475.
7. Löhner R. Robust, vectorized search algorithms for interpolation on unstructured grids // J. of Computational Physics. 1995. **118**, N 2. 380–387.
8. Apte S.V., Mahesh K., Moin P., Oefelein J.C. Large-eddy simulation of swirling particle-laden flows in a coaxial-jet combustor // Int. J. of Multiphase Flow. 2003. **29**, N 8. 1311–1331.
9. Zhou Q., Leschziner M.A. An improved particle-locating algorithm for Eulerian–Lagrangian computations of two-phase flows in general coordinates // Int. J. of Multiphase Flow. 1999. **25**, N 5. 813–825.
10. Скворцов А.В. Обзор алгоритмов построения триангуляции Делоне // Вычислительные методы и программирование. 2002. **3**, № 1. 18–43.
11. Chen X.-Q., Pereira J.C.F. A new particle-locating method accounting for source distribution and particle-field interpolation for hybrid modeling of strongly coupled two-phase flows in arbitrary coordinates // Numerical Heat Transfer. Part B. 1999. **35**, N 1. 41–63.
12. Chorda R., Blasco J.A., Fueyo N. An efficient particle-locating algorithm for application in arbitrary 2D and 3D grids // Int. J. of Multiphase Flow. 2002. **28**, N 9. 1565–1580.
13. Cheng H.-P., Cheng J.-R., Yeh G.-T. A particle tracking technique for the lagrangian-eulerian finite element method in multi-dimensions // Int. J. for Numerical Methods in Engineering. 1996. **39**, N 7. 1115–1136.

Поступила в редакцию
14.01.2008