

УДК 532.529

## РЕАЛИЗАЦИЯ МЕТОДА КОНЕЧНЫХ ОБЪЕМОВ И РАСЧЕТ ТЕЧЕНИЙ ВЯЗКОГО СЖИМАЕМОГО ГАЗА НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ

К. Н. Волков<sup>1</sup>, В. Н. Емельянов<sup>1</sup>, А. Г. Карпенко<sup>2</sup>, П. Г. Смирнов<sup>1</sup>, И. В. Тетерина<sup>1</sup>

Обсуждаются возможности использования графических процессоров общего назначения для моделирования течений вязкого сжимаемого газа. Для дискретизации уравнений Навье–Стокса используется метод конечных объемов на неструктурированной сетке. Для программной реализации параллельных вычислительных алгоритмов применяется технология CUDA. Приводится решение ряда модельных задач газовой динамики на графических процессорах и обсуждаются подходы к оптимизации программного кода, связанные с использованием различных типов памяти. Сравнивается ускорение счета на графических процессорах по отношению к расчетам на центральном процессоре при использовании сеток различной разрешающей способности и при различных способах разбиения исходных данных на блоки.

**Ключевые слова:** графический процессор, параллельный алгоритм, газовая динамика, метод конечных объемов, неструктурированная сетка, технология CUDA.

**1. Введение.** Перенос вычислений с центральных процессоров (ЦПУ) на графические процессоры (ГПУ) связан не с визуализацией обрабатываемых данных, а с необходимостью ускорения работы программ, предназначенных, например, для моделирования течений жидкости и газа. Область использования ГПУ расширяется благодаря появлению новых технологий программирования и усовершенствованию аппаратного обеспечения. Основным критерием эффективности применения ГПУ является скорость вычислений, измеряемая количеством операций в секунду, и ускорение решения задачи на ГПУ по сравнению с вычислениями на ЦПУ.

Некоторые возможности использования графических процессоров общего назначения для численного решения задач динамики вязкой несжимаемой жидкости обсуждаются в работе [1]. В частности, рассматриваются особенности реализации схемы расщепления (метод проекции) и многосеточного метода, предназначенного для решения системы разностных уравнений, порожденных конечно-разностной дискретизацией уравнения Пуассона для давления.

Дву- и трехмерные уравнения Эйлера решаются в работе [2]. Для программной реализации используется язык BrookGPU (видеокарта ATI 1950XT) и технология CUDA (видеокарта NVIDIA GeForce 8800 GTX). При использовании языка BrookGPU двумерные расчеты дают ускорение примерно в 29 раз по сравнению с расчетами на ЦПУ, а трехмерные расчеты — примерно в 3 раза. Применение технологии CUDA для распараллеливания трехмерного кода позволяет увеличить производительность вычислений до 16 раз за счет эффективного использования различных типов памяти.

На неструктурированной сетке с ячейками в виде тетраэдров трехмерные уравнения Эйлера решаются в работах [3, 4].

Реализация кода [3] проводится таким образом, что на каждый контрольный объем приходится одна нить. Каждая нить считывает объем ячейки и консервативные переменные из глобальной памяти, после чего находятся физические переменные и рассчитывается их вклад в потоки через грани контрольного объема. Проводится цикл по всем граням контрольного объема и рассчитываются невязкие и вязкие потоки при помощи суммирования вкладов всех граней. Производится считывание нормалей к граням контрольного объема и индексов соседних контрольных объемов, которые используются для доступа к соответствующим им консервативным переменным. На последнем шаге невязкие и вязкие потоки применяются для вычисления невязки. Такой подход требует излишних действий по расчету потоков и связанных с ними функций.

<sup>1</sup> Балтийский государственный технический университет “Военмех” им. Д. Ф. Устинова, факультет энергетического машиностроения, 1-я Красноармейская ул., д. 1, 190005, Санкт-Петербург; К. Н. Волков, вед. науч. сотр., e-mail: dsci@mail.ru; В. Н. Емельянов, профессор, e-mail: vlademelyanov@gmail.com; П. Г. Смирнов, аспирант, e-mail: petr.s.8314@mail.ru; И. В. Тетерина, доцент, e-mail: yaiv@mail.ru

<sup>2</sup> Санкт-Петербургский государственный университет, математико-механический факультет, Петродворец, Университетский просп., 28, 198504, Санкт-Петербург; аспирант, e-mail: aspera.2003.ru@mail.ru

Модифицированный подход состоит в том, чтобы сначала рассчитать потоки через грани всех контрольных объемов, избегая повторных вычислений [4]. При этом требуется несколько операций (три операции для контрольного объема в виде тетраэдра) по чтению из глобальной памяти вкладов граней контрольного объема в потоки вместо однократного чтения консервативных переменных. При использовании неструктурированных сеток соседние контрольные объемы хранятся в памяти ГПУ далеко друг от друга, что увеличивает время загрузки данных в регистры и снижает итоговую производительность. Для повышения эффективности расчетов используется схема переиндексации контрольных объемов (*renumbering scheme*), которая гарантирует, что соседние контрольные объемы располагаются в смежных ячейках памяти, а это упрощает и убыстряет доступ к ним из программного кода.

Возможности подходов, рассмотренных в [3 и 4], демонстрируются на примере моделирования сверхзвукового обтекания профиля NASA0012 при нулевом угле атаки. Ускорение вычислений на ГПУ составляет около 9.5 раз относительно ускорения, достигнутого при использовании параллельного кода на четырехъядерном процессоре (применяется технология OpenMP), и около 33 относительно расчетов на основе последовательной версии кода. Расчеты проводятся на платформе NVIDIA Tesla C2050 и центральном процессоре Intel Core 2 Q9450.

Трехмерные расчеты на блочно-структурированных сетках рассматриваются в работе [5] применительно к моделированию турбулентных течений в межлопаточных каналах газовых турбин на основе метода конечных объемов и модели турбулентности Прандтля. Для ускорения сходимости к стационарному решению используются локальный шаг по времени и многосеточный метод. Расчеты с использованием явной схемы второго порядка по времени и различного числа узлов ГПУ-кластера (максимальное число узлов составляет 16) показывают практически линейный прирост ускорения при увеличении числа узлов. Достигнутое ускорение составляет около 19 относительно расчетов на двухъядерном ЦПУ (используется видеокарта NVIDIA GT200).

Сжимаемые уравнения Навье–Стокса, замкнутые при помощи модели турбулентности Спаларта–Аллмареса, решаются в работе [6] на дву- и трехмерных неструктурированных сетках (ячейки сетки представляют собой треугольники и тетраэдры). Дискретизация по времени проводится при помощи неявной схемы. Для дискретизации невязких потоков применяется схема Рунге. Дискретизация вязких потоков основывается на использовании примитивных переменных и их градиентов. Расчеты проводятся с одинарной (ускорение составляет 28), двойной (ускорение составляет 19.6) и смешанной (ускорение составляет 25.2) точностью. Расчеты с двойной и смешанной точностью позволяют получить более низкий уровень невязки и более точное распределение давления по поверхности профиля по сравнению с расчетами с одинарной точностью. Для расчетов используется видеокарта NVIDIA GeForce GTX 280 и технология CUDA. Последовательный код запускается на процессоре Intel Core 2 Duo с тактовой частотой 2.8 ГГц. Разрабатывается двухуровневый подход к оптимизации формы профиля. На уровне 1 используются вычисления с одинарной точностью (грубое решение), а на уровне 2 — вычисления со смешанной точностью (уточнение решения).

Влияние порядка точности разностной схемы на ускорение вычислений на ГПУ обсуждается в работе [7] (рассматриваются схемы 3-го, 5-го, 7-го и 9-го порядков). Производительность вычислений с одинарной и двойной точностью увеличивается на два порядка по сравнению с расчетами на ЦПУ и оказывается выше для схем высокого порядка точности при прочих одинаковых условиях. Расчеты проводятся на платформе Tesla C1060 и кластере, содержащем 856 процессоров.

В настоящей статье рассматривается реализация метода конечных объемов, предназначенного для моделирования нестационарных течений вязкого сжимаемого газа на неструктурированных сетках, при использовании графических процессоров общего назначения. Обсуждается решение ряда модельных задач газовой динамики, имеющиеся возможности по оптимизации программного кода, а также ускорение вычислительной процедуры, полученное на сетках различной разрешающей способности и при различных способах распределения данных по блокам, по сравнению с расчетами на центральном процессоре.

**2. Метод конечных объемов.** Для дискретизации уравнений Навье–Стокса на неструктурированной сетке используется метод конечных объемов.

**2.1. Основные уравнения.** В декартовой системе координат  $(x, y, z)$  нестационарное течение вязкого сжимаемого газа описывается следующим уравнением, записанным в консервативных переменных (по повторяющимся индексам предполагается суммирование):

$$\frac{\partial Q}{\partial t} + \frac{\partial F_i}{\partial x_i} = 0. \quad (1)$$

Уравнение состояния совершенного газа представляется в форме  $p = (\gamma - 1)\rho \left[ e - \frac{1}{2}(v_x^2 + v_y^2 + v_z^2) \right]$ . Вектор

консервативных переменных  $Q$  и вектор потока  $F_i$  имеют следующий вид:

$$Q = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ \rho e \end{pmatrix}, \quad F_i = \begin{pmatrix} \rho v_i \\ \rho v_i v_x - \tau_{ix} + p\delta_{ix} \\ \rho v_i v_y - \tau_{iy} + p\delta_{iy} \\ \rho v_i v_z - \tau_{iz} + p\delta_{iz} \\ (\rho e + p)v_i - \tau_{ix}v_x - \tau_{iy}v_y - \tau_{iz}v_z + q_i \end{pmatrix}.$$

Здесь  $t$  — время;  $\rho$  — плотность;  $v_x, v_y, v_z$  — составляющие скорости в координатных направлениях  $x, y$  и  $z$ ;  $e$  — полная энергия единицы массы;  $p$  — давление;  $\gamma$  — отношение удельных теплоемкостей.

Компоненты тензора вязких напряжений и вектора теплового потока находятся из соотношений

$$\tau_{ij} = \mu \left( \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} - \frac{2}{3} \frac{\partial v_k}{\partial x_k} \delta_{ij} \right), \quad q_i = -\lambda \frac{\partial T}{\partial x_i},$$

где  $T$  — температура. Вязкость зависит от температуры и находится из закона Сазерленда. Теплопроводность выражается через вязкость и число Прандтля, которому присваивается постоянное значение. При моделировании турбулентных течений молекулярные коэффициенты переноса заменяются их эффективными значениями [8].

**2.2. Особенности дискретизации.** Интегрируя уравнение (1) по контрольному объему  $V_i$  с границей  $\partial V_i$  и применяя теорему Гаусса–Остроградского, получим

$$\frac{d}{dt} \int_{V_i} Q d\Omega + \oint_{\partial V_i} F_i dS = 0.$$

Предположим, что сеточная величина  $Q_i$ , определенная в центре контрольного объема  $V_i$ , представляет собой среднее интегральное значение соответствующей непрерывно распределенной величины  $Q_i = \frac{1}{V_i} \int_{V_i} Q d\Omega$ . Вычисляя интеграл по границе контрольного объема как сумму произведений значений вектора потока в центрах граней контрольного объема  $F_i$  на площади его граней  $S_i$ , получим  $\frac{dQ_i}{dt} = R_i(Q, \nabla Q)$ . Невязка рассчитывается при помощи суммирования потоков через грани контрольных объемов:  $R_i(Q, \nabla Q) = -\frac{1}{V_i} \sum_{j=1}^{N_i} F_{ij} S_{ij}$ , где  $F_{ij}$  — поток через грань  $(i, j)$  (определяется в центре грани),  $S_{ij}$  — площадь грани  $(i, j)$ ,  $N_i$  — число граней контрольного объема  $i$ .

Для дискретизации по времени применяется явный двухшаговый метод Рунге–Кутты (модифицированный метод Эйлера)  $Q_i^* = Q_i^n + \Delta t R_i(Q^n)$ ,  $Q_i^{n+1} = Q_i^* + \Delta t R_i(Q^*)$ , где  $\Delta t$  — шаг по времени. Индекс  $n$  соответствует слою по времени.

Различные версии метода конечных объемов отличаются друг от друга способом вычисления потоков через грани контрольного объема [9]. Расчет потоков основывается на решении задачи о распаде произвольного разрыва (применяются метод Годунова и схема Рое).

Для расчета невязких потоков используется свойство вращательной симметрии уравнений Эйлера, при этом  $\mathbf{F} \cdot \mathbf{a} = \tilde{L} F_x (\tilde{L}^{-1} Q)$ , где  $\mathbf{a}$  — произвольный единичный вектор, матрица  $\tilde{L} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 1 \end{pmatrix}$ . Матрица  $L^{-1}$  является матрицей вращения системы координат, при этом вектор  $\mathbf{e}'_1$  новой системы координат совпадает с вектором  $\mathbf{a}$  ( $\mathbf{e}'_1 = \mathbf{a}$ ), а новая система координат остается ортогональной ( $\mathbf{e}'_3 = \mathbf{e}'_1 \times \mathbf{e}'_2$ ), причем  $\mathbf{e}'_2 = \mathbf{e}_k \times \mathbf{a}$ , где  $k = 1, 2, 3$ . Под  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  понимается ортонормированный базис основной системы координат. При  $\mathbf{a} = \mathbf{n}$  поток тензорного поля через поверхность контрольного объема находится из соотношения  $\int_{\partial V_i} (\mathbf{F} \cdot \mathbf{n}) dS = \tilde{L} \int_{\partial V_i} F_x (\tilde{L}^{-1} Q) dS$ .

Для расчета потоков в многомерном случае консервативные переменные преобразуются в примитивные, а задача Римана решается для вектора  $V = \tilde{L}^{-1} Q$ . Вектор  $\mathbf{e}'_2$  выбирается таким образом, чтобы угол между образующими его векторами  $\mathbf{e}_k$  и  $\mathbf{n}$  был максимальным (это позволяет уменьшить влияние ошибок округления).

**2.3. Расчет потоков.** Схема Рое [10] основана на квазилинейной форме системы уравнений Эйлера и предполагает построение матрицы, удовлетворяющей ряду условий, при которых задача становится гиперболической и разрешимой, а метод — консервативным. Пренебрегая вкладом вязких слагаемых, уравнение (1) можно записать в форме

$$\frac{\partial Q}{\partial t} + \tilde{A}(Q_L, Q_R) \frac{\partial F(Q)}{\partial x} = 0. \quad (2)$$

Поток через грань контрольного объема рассчитывается по формуле

$$F_{i+1/2} = \frac{1}{2} [F(Q_R) + F(Q_L)] - \frac{1}{2} \left| \tilde{A}_{i+1/2} \right| (Q_R - Q_L). \quad (3)$$

Считается, что  $\Delta F = \tilde{A}_{i+1/2}(Q_L, Q_R) \Delta Q$  для любых векторов  $Q_L$  и  $Q_R$  и  $\tilde{A}_{i+1/2}(Q, Q) = \tilde{A}_{i+1/2}(Q)$ . Все собственные значения матрицы  $\tilde{A}_{i+1/2}(Q_L, Q_R)$  являются действительными, и существует полный набор собственных векторов.

Пусть  $v$  — модуль скорости. Введем матрицу перехода между консервативными  $Q$  и примитивными  $V$

переменными: 
$$P = \frac{\partial Q}{\partial V} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ v_x & \rho & 0 & 0 & 0 \\ v_y & \rho & 0 & 0 & 0 \\ v_z & \rho & 0 & 0 & 0 \\ v^2/2 & \rho v_x & \rho v_y & \rho v_z & 1/(\gamma - 1) \end{pmatrix}.$$

В примитивных переменных уравнение (2) примет вид  $\frac{\partial V}{\partial t} + P^{-1} A_x P \frac{\partial F_x(V)}{\partial x} = 0.$

Якобиан в примитивных переменных ( $A^x = P^{-1} A_x P$ ) записывается как 
$$A^x = \begin{pmatrix} v_x & \rho & 0 & 0 & 0 \\ 0 & v_x & 0 & 0 & 1/\rho \\ 0 & 0 & v_x & 0 & 0 \\ 0 & 0 & 0 & v_x & 0 \\ 0 & \rho c^2 & 0 & 0 & v_x \end{pmatrix}.$$

Собственные значения якобиана находятся из уравнения  $\det(A^x - \lambda I) = 0$ , решение которого дает  $\lambda_1 = v_x$ ,  $\lambda_2 = v_x$ ,  $\lambda_3 = v_x$ ,  $\lambda_4 = v_x - c$ ,  $\lambda_5 = v_x + c$ .

Введем матрицу  $R$ , столбцами которой являются правые собственные векторы матрицы  $A^x$ , полученные из решения уравнения  $(A^x - \lambda I)r_i = 0$ . Якобиан в примитивных переменных представляется в виде  $A^x = R \Lambda R^{-1}$ , где  $\Lambda = \text{diag} \{ \lambda_i \}$  — матрица собственных значений и  $R$  — матрица правых собственных векторов. Якобиан в консервативных переменных имеет вид  $A_x = M \Lambda M^{-1}$ , где  $M$  — матрица правых собственных векторов ( $M = P R$ ). Соотношение (3) для расчета потока примет вид

$$F_{i+1/2} = \frac{1}{2} [F(Q_R) + F(Q_L)] - \frac{1}{2} M_{i+1/2} |\Lambda_{i+1/2}| M_{i+1/2}^{-1} (Q_R - Q_L).$$

Коэффициенты матриц  $M_{i+1/2}$  и  $\Lambda_{i+1/2}$  вычисляются через осредненные по Рое значения примитивных переменных (черта соответствует осреднению)  $\bar{\rho}_{i+1/2} = (\rho_{i+1} \rho_i)^{1/2} = R_{i+1/2} \rho_i$ ;

$$\bar{u}_{i+1/2} = \frac{(u \rho^{1/2})_{i+1} + (u \rho^{1/2})_i}{\rho_{i+1}^{1/2} + \rho_i^{1/2}} = \frac{R_{i+1/2} u_{i+1} + u_i}{R_{i+1/2} + 1}; \quad \bar{H}_{i+1/2} = \frac{(H \rho^{1/2})_{i+1} + (H \rho^{1/2})_i}{\rho_{i+1}^{1/2} + \rho_i^{1/2}} = \frac{R_{i+1/2} H_{i+1} + H_i}{R_{i+1/2} + 1}.$$

Здесь  $H$  — полная энтальпия.

Диссипативный член в соотношении (3) задается в форме  $\frac{1}{2} \left| \tilde{A}_{i+1/2} \right| (Q_{i+1} - Q_i) = \frac{1}{2} \sum_j |\lambda_j| \Delta W r_j$ , где

$r_j$  — столбцы матрицы  $M_{i+1/2}$ . Приращение характеристических переменных  $\Delta W = M_{i+1/2}^{-1} (Q_R - Q_L)$

выражается через приращения примитивных переменных: 
$$\Delta W = \begin{pmatrix} \Delta \rho - \Delta p / c^2 \\ \Delta v_z \\ -\Delta v_y \\ \Delta p / (\rho c) - \Delta v_x \\ \Delta p / (\rho c) + \Delta v_x \end{pmatrix}.$$

Недостаток подхода состоит в отсутствии механизма различения волн разрежения и сжатия. При обращении в нуль собственных значений вклад соответствующих волн в приращение вектора потока становится нулевым. В случае волны разрежения, которая, в отличие от волны сжатия и контактного разрыва, занимает определенную область пространства, это приводит к получению нефизического решения (скачок разрежения). Проблема получения нефизического решения связана с условием энтропии, которое предполагает выполнение дискретного аналога энтропийного неравенства для разностной схемы. В разностную схему вблизи звуковой точки вводится искусственная вязкость [11]

$$|\lambda_p^*| = \begin{cases} |\lambda_p| & \text{при } |\lambda_p| \geq \varepsilon_p, \\ \frac{1}{2} \left( \frac{\lambda_p^2}{\varepsilon_p} + \varepsilon_p \right) & \text{при } |\lambda_p| < \varepsilon_p, \end{cases}$$

где  $\varepsilon_p = 2 \max\{\Delta\lambda_p, 0\}$ . Значения  $p = 4$  и  $p = 5$  соответствуют собственным значениям  $\lambda_4 = v_x - c$  и  $\lambda_5 = v_x + c$ .

Ограничение на шаг по времени следует из условия, что волны, образовавшиеся после распада разрыва на грани контрольного объема, не достигли его центра, а при более слабом ограничении — другой грани. Схема Годунова обеспечивает положительность решения, если  $CFL = 2|\lambda_{\max}| \frac{\Delta t}{\Delta r} < 1$ , где  $\Delta r$  — расстояние от центра ячейки до ближайшей грани и CFL означает условие Courant–Friedrichs–Lewy.

**2.4. Повышение точности.** Для повышения порядка точности по пространству используется кусочно-непрерывное распределение искомых функций внутри контрольного объема и производится интерполяция решения на грань контрольного объема  $F_{i+1/2} = F_{i+1/2}(Q_L^*, Q_R^*)$ . В схему вводятся переключатели, которые осуществляют переход на схему первого порядка в тех областях, где существуют большие градиенты решения, и на схему второго порядка в области гладкого изменения решения:  $Q_L^* = Q_L + \overline{\nabla Q} |\Delta \mathbf{r}_L|$ ,  $Q_R^* = Q_R - \overline{\nabla Q} |\Delta \mathbf{r}_R|$ . Функция limiter используется для ограничения значений искомых функций при интерполяции:  $\overline{\nabla Q} = \text{limiter} \left\{ (\nabla Q)_L \cdot \frac{\Delta \mathbf{r}_L}{|\Delta \mathbf{r}_L|}, (\nabla Q)_R \cdot \frac{\Delta \mathbf{r}_R}{|\Delta \mathbf{r}_R|} \right\}$ . Под  $\Delta \mathbf{r}_L$  и  $\Delta \mathbf{r}_R$  понимаются векторы, направленные от центров соседних контрольных объемов  $L$  и  $R$  к серединной точке разделяющей их грани.

Интегрируя по поверхности контрольного объема и выбирая за значение функции на грани среднее из узловых значений, получим градиент в центре ячейки:

$$(\nabla Q)_i^l = \frac{1}{V_i} \sum_{k=1}^{N_i} \frac{1}{2} (Q_i + Q_k) n_l S_{ik}, \quad l = 1, 2, 3.$$

Для расчета вязких потоков проводится их линеаризация, причем для простоты якобиан рассчитывается в примитивных переменных. Градиент в центре грани контрольного объема рассчитывается при помощи линейной интерполяции градиента на грань:

$$\nabla Q = (\nabla Q)_L + \frac{|\Delta \mathbf{r}_L|}{|\Delta \mathbf{r}_L| + |\Delta \mathbf{r}_R|} [(\nabla Q)_R - (\nabla Q)_L].$$

Некоторые детали, связанные с расчетом вязких потоков, приводятся в работе [9].

**3. Особенности реализации.** Для ГПУ требуется разработка алгоритмов, имеющих высокую степень параллелизма на уровне данных. При этом одна операция выполняется над всеми элементами массива данных (под элементом массива понимается структура данных или несколько чисел, хранящихся во внешней памяти).

**3.1. Схема решения задачи.** Формирование контрольных объемов и расчет их геометрических характеристик производится на ЦПУ (рис. 1). На ГПУ выполняются операции, связанные с вычислением потоков и расчетом шага интегрирования по времени.

Ядро 1 выполняет расчет потоков через грани контрольных объемов (расчеты подобного типа основываются на использовании двух узлов  $i$  и  $j$ , соединенных гранью). Расчет локального шага интегрирования по времени выполняется в ядре 2 для каждого контрольного объема. Поиск минимального шага по времени осуществляется в ядре 3, которое производит перебор всех контрольных объемов сетки. Продвижение решения во времени выполняет ядро 4. Ядра 1, 2 и 4 обладают высокой степенью параллелизма, а ядро 3 требует коммуникаций между блоками, снижая общую производительность вычислений.

**3.2. Разбиение данных.** При решении задач газовой динамики на многопроцессорных вычислительных системах используется принцип геометрического параллелизма, который предполагает геометрическую декомпозицию расчетной области на подобласти, содержащие приблизительно равное число узлов

сетки. На практике находят применение различные подходы к декомпозиции, предполагающие разбиение расчетной области вдоль одного, двух или трех координатных направлений. В соответствии с типом декомпозиции возникает три различных подхода к распараллеливанию, а также три типа топологии межпроцессорных связей (линейка, квадратная решетка и кубическая решетка).

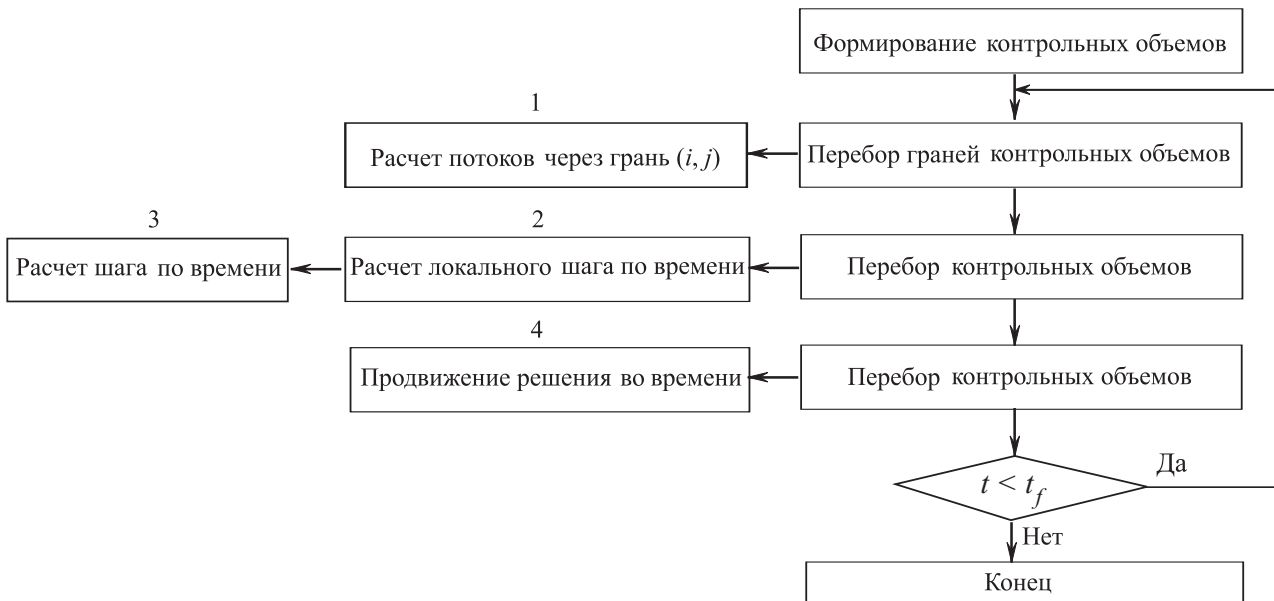


Рис. 1. Схема решения задачи

Для реализации явной схемы при линейном разбиении области вдоль одного координатного направления (например, по координате  $x_2$ ) процессор  $k$  производит вычисления на верхнем слое по времени в диапазоне узлов

$$I^{(k)} = \left\{ (i_1, i_2, i_3), i_1 = 0, \dots, N_1, i_2 = i_{20}^{(k)}, \dots, i_{21}^{(k)}, i_3 = 0, \dots, N_3 \right\} \equiv I_1 \times I_2^k \times I_3.$$

Для получения равномерной загрузки процессоров количество точек  $N^{(k)} = N_1 (i_{21}^{(k)} - i_{20}^{(k)} + 1) N_3$  в диапазоне  $I^{(k)}$  должно быть примерно одинаковыми.

При использовании различных методов декомпозиции эффективность вычислительной процедуры зависит от схемы обменов и числа процессоров. При прочих равных условиях использование линейной топологии при большом числе процессоров является менее эффективным, чем использование топологии типа решетки (размер передаваемых данных снижается с ростом числа процессоров). В качестве многопроцессорной системы выступает кластер, содержащий несколько ГПУ, каждый из которых рассчитывает свою подобласть.

Явная схема обладает высокой степенью параллелизма, поскольку значения на новом временном слое рассчитываются в каждом узле независимо друг от друга, при этом отпадает необходимость в использовании циклических конструкций. Каждому узлу сетки соответствует своя нить ГПУ, которая производит расчеты параллельно с другими нитями.

**3.3. Размещение векторов в памяти.** При реализации метода конечных объемов каждому контрольному объему соответствует пять физических переменных (плотность, три компоненты скорости и давление). В расчетах на ЦПУ данные развертываются в линейный массив путем записи  $n_e$  блоков по пять значений. Адресация к искомой функции с порядковым номером  $j$  в контрольном объеме  $i$  имеет вид  $u[5 * i + j]$ . Для ГПУ более эффективный способ адресации состоит в том, чтобы развернуть данные в пять блоков по  $n_e$  значений, что соответствует адресации  $u[i + j * n_e]$ . Такое размещение векторов в памяти соответствует модели объединения запросов, что приводит к увеличению производительности кода.

Оптимизация производительности достигается за счет уменьшения числа обращений к глобальной памяти в каждой нити. Расчет левой и правой частей дискретного уравнения в каждом узле сетки основывается на переборе всех граней, связанных с данным узлом, и суммировании вкладов потоков через эти грани. Минимизация числа обращений к глобальной памяти достигается через использование разделяемой памяти и при помощи обновления значений переменных, хранящихся в глобальной памяти, по окончании всех расчетов.

При реализации разностных схем расчета потоков данные, требуемые на шаге эволюции, рассчитываются перед началом шага по времени и хранятся в текстурной памяти. Результаты, полученные на шаге по времени, хранятся в глобальной памяти. Вместо обменов данными между соседними блоками, которые соответствуют фиктивным ячейкам, проводится расчет параметров в фиктивных ячейках каждого блока.

**3.4. Расчет потоков.** Реализацию метода конечных объемов с использованием глобальной памяти и файла регистров поясняет рис. 2. На каждом слое по времени расчеты выполняются в два этапа. Для параллельной реализации метода конечных объемов на ГПУ используется два ядра, одно из которых производит вычисление потоков через грани контрольных объемов (этап 1, фрагмент рис. 2а), а другое осуществляет продвижение решения во времени (этап 2, рис. 2б).

На этапе 1 в глобальной памяти в массиве  $Q$  находятся значения переменных в центре контрольного объема. Расчет потока через каждую грань контрольного объема производится своей нитью. Каждая нить использует значения переменных в контрольных объемах  $i$  и  $i + 1$ . Потоки через грани контрольного объема сохраняются в массиве  $F$ .

На этапе 2 каждому контрольному объему соответствует своя нить. Для нахождения искомым функций на новом слое по времени используются потоки через грани  $i - 1/2$  и  $i + 1/2$  и решение в контрольном объеме  $i$ . Решение сохраняется в массиве  $Q$ .

**3.5. Организация кода.** На практике используется два подхода к расчету потоков, в одном из которых потоки рассчитываются в начале счета и сохраняются в отдельном массиве (pre-compute), а в другом — расчет потоков производится на каждом шаге в процессе счета (re-compute) [3, 4].

Расчет потоков через грани контрольного объема требует доступа к переменным, соответствующим соседним контрольным объемам (рис. 3). Такой доступ выполняется наиболее быстро, если соседние контрольные объемы располагаются в памяти рядом друг с другом. Для объединения запросов и ускорения вычислений на неструктурированных сетках используется переиндексация контрольных объемов.

Одна нить осуществляет обработку одного контрольного объема. Локальную индексацию узлов поясняет рис. 4. При переборе узлов или граней контрольного объема (рис. 4а) производится их переиндексация таким образом, чтобы узлы, принадлежащие одному контрольному объему, располагались в памяти по соседству (рис. 4б). Подход позволяет использовать механизм объединения запросов к памяти (рис. 4в).

Для того чтобы избежать обновления узловых значений искомым функций при переборе граней контрольных объемов, используются различные методы (рис. 5). В методе 1 обновление узловых значений производится только одной из выбранных граней. Метод 2 основан на такой расцветке граней, при которой никакая из двух граней одинакового цвета не производила расчеты в том же самом узле. В методе 3 применяются атомарные операции, позволяющие произвести операции чтения, сложения и записи за одну операцию. Одинаковые индексы соответствуют граням одинакового цвета.

Использование разделяемой памяти при расчете потоков поясняет рис. 6, который показывает порядок копирования данных из глобальной памяти в разделяемую память. Например, реализация схемы типа MUSCL (Monotonic Upwind Scheme for Conservation Laws) требует использования трех контрольных объемов для расчета потоков и их ограничителей. На шаге 1 производится копирование векторов решения, соответствующих центрированному расположению (рис. 6а), а на шагах 2 и 3 — копирование векторов решения, находящихся слева и справа (рис. 6б и рис. 6в). Каждая нить производит обработку

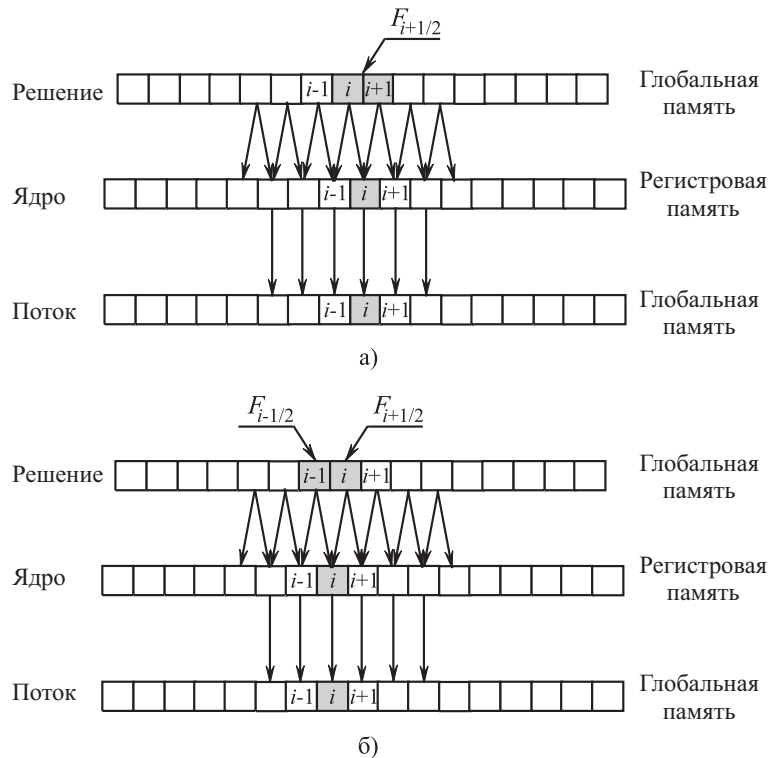


Рис. 2. Расчет потоков (а) и расчет значений искомой функции на новом слое по времени (б)

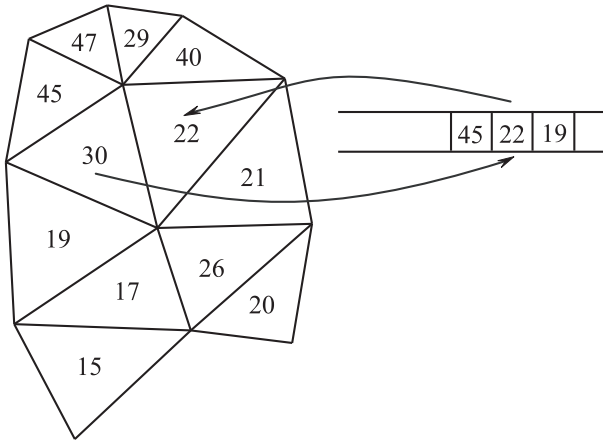


Рис. 3. Расчет потоков

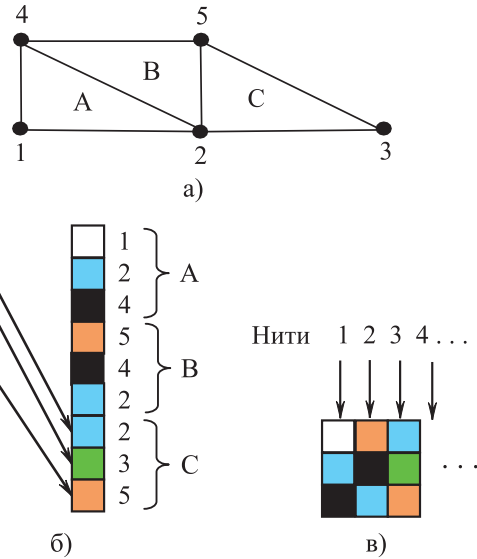


Рис. 4. Локальная индексация узлов, позволяющая обеспечить объединение запросов к памяти

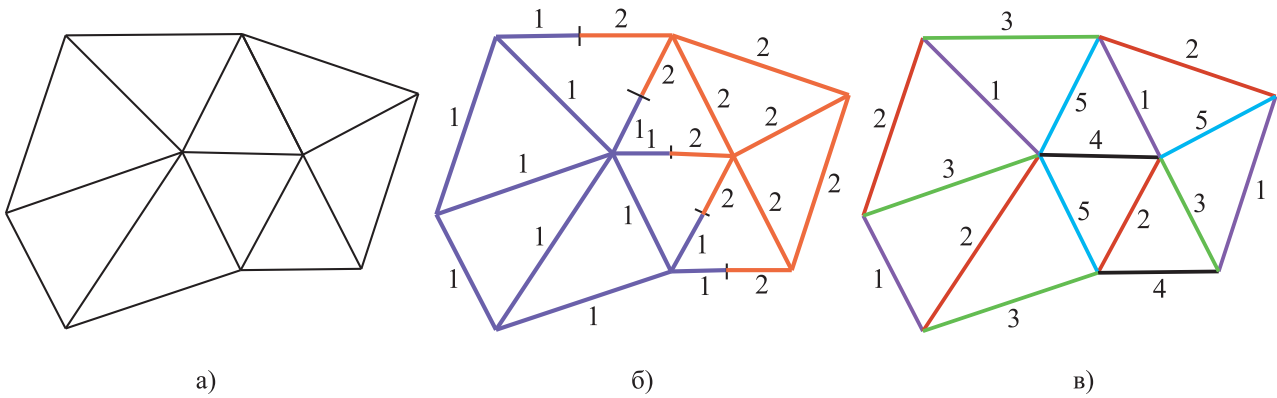


Рис. 5. Расчет узловых значений: сетка (а), метод 1 (б), метод 2 (в). Цифры соответствуют граням различного цвета

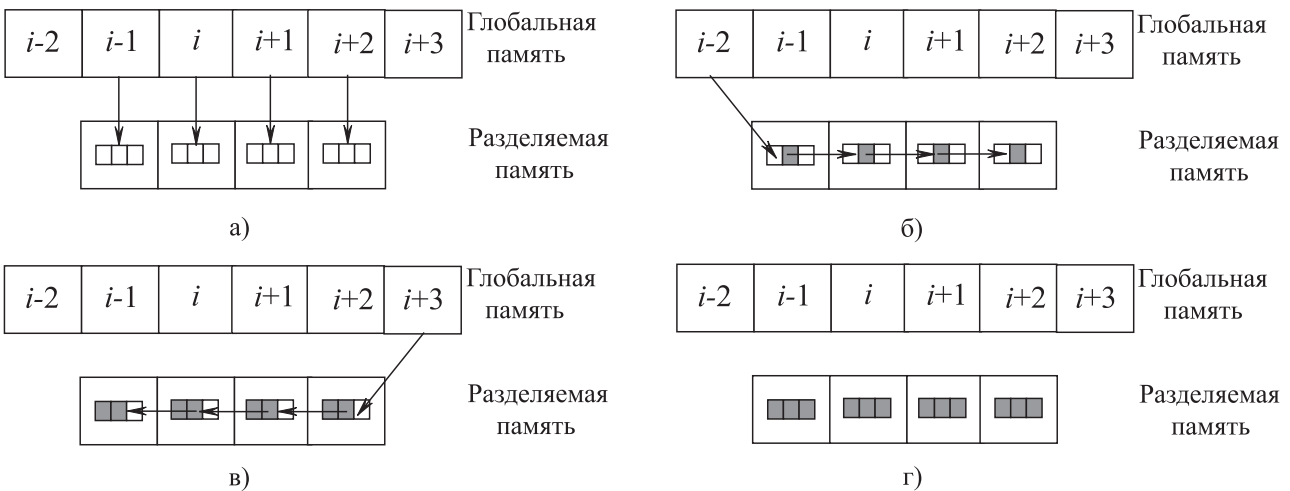


Рис. 6. Использование разделяемой памяти при расчете потоков

трех векторов решения, находящихся в разделяемой памяти (рис. 6г).

Порядок обновления переменных в разделяемой памяти показывает табл. 1. При моделировании двумерных течений (четыре переменные) каждая нить производит обработку четырех векторов решения, а



общий объем используемой разделяемой памяти составляет  $4 * n_v * \text{blockDim.y} * \text{blockDim.x}$ .

Таблица 1  
Порядок обновления переменных и расчета потоков

| Шаг            | Нить  |                                   |   |
|----------------|---|-----------------------------------|---|
|                | $i - 1$                                     | $i$                               | $i + 1$                                     |
| Переменные     | $u_{i-2}, u_{i-1}, u_i$                     | $u_{i-1}, u_i, u_{i+1}$           | $u_i, u_{i+1}, u_{i+2}$                     |
| Реконструкция  | $u_{i-1}^L, u_{i-1}, u_{i-1}^R$             | $u_i^L, u_i, u_i^R$               | $u_{i+1}^L, u_{i+1}, u_{i+1}^R$             |
| Ограничение    | $u_{i-1}^{L*}, u_{i-1}, u_{i-1}^{R*}$       | $u_i^{L*}, u_i, u_i^{R*}$         | $u_{i+1}^{L*}, u_{i+1}, u_{i+1}^{R*}$       |
| Эволюция       | $\hat{u}_{i-1}^L, u_{i-1}, \hat{u}_{i-1}^R$ | $\hat{u}_i^L, u_i, \hat{u}_i^R$   | $\hat{u}_{i+1}^L, u_{i+1}, \hat{u}_{i+1}^R$ |
| —              | Синхронизация                               |                                   |   |
| Функция потока | $f(\hat{u}_{i-2}^R, \hat{u}_{i-1}^L)$       | $f(\hat{u}_{i-1}^R, \hat{u}_i^L)$ | $f(\hat{u}_i^R, \hat{u}_{i+1}^L)$           |
| —              | Синхронизация                               |                                   |   |
| Поток          | $f_{i-2} - f_{i-1}$                         | $f_{i-1} - f_i$                   | $f_i - f_{i+1}$                             |

**4. Задача Сода.** Рассмотрим решение задачи Сода [12], предназначенной для сравнения различных схем расчета потоков (рис. 7). В результате распада произвольного разрыва влево распространяется волна разрежения, а вправо — ударная волна и контактный разрыв.

Дискретизация уравнений Эйлера проводится при помощи метода конечных объемов. Для явной дискретизации по времени используются схема Эйлера и двухшаговая схема Рунге–Кутты. Расчет потоков через грани контрольных объемов производится при помощи схем первого или второго порядка.

Задача о распаде произвольного разрыва решается точным итерационным методом Годунова или приближенным методом Слева от разделяющей перегородки полагается, что  $\rho_L = 1, u_L = 0.75, p_L = 1$ , а справа —  $\rho_R = 0.125, u_R = 0, p_R = 1$ . Расчеты проводятся на сетках с различным числом ячеек. При переходе от одной сетки к другой число ячеек увеличивается в несколько раз (1024 ячеек для сетки 1, 30720 ячеек для сетки 2 и 307200 ячеек для сетки 3). Сетка наилучшей разрешающей способности содержит около трех миллионов ячеек (сетка 4). Шаг по времени составляет  $1.52 \times 10^{-5}$  с, а общее время счета —  $7.63 \times 10^{-3}$  с. Число Куранта равняется 0.85. Расчеты проводятся на одном модуле платформы Tesla S1070 с тактовой частотой ядра 1.44 ГГц (количество ядер составляет 256), а также на одном ядре центрального процессора AMD Phenom 2 с тактовой частотой 3 ГГц.

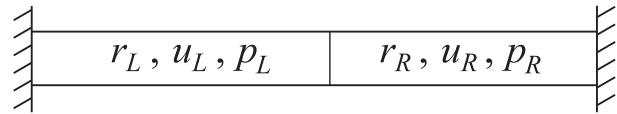


Рис. 7. Задача Сода

Время, необходимое для расчета одного шага по времени, а также ускорение решения задачи приводятся в табл. 2 (время указывается в миллисекундах). Вариант 1 соответствует расчету по схеме Годунова, а вариант 2 — расчету по схеме Рое. В обоих вариантах расчета наблюдается хороший прирост ускорения  $S$  при увеличении числа контрольных объемов сетки.

Задача Сода. Время и ускорение вычислений

Таблица 2

| № | Сетка 1 |      |       | Сетка 2 |      |        | Сетка 3 |      |        | Сетка 4 |      |        |
|---|---------|------|-------|---------|------|--------|---------|------|--------|---------|------|--------|
|   | ЦПУ     | ГПУ  | $S$   | ЦПУ     | ГПУ  | $S$    | ЦПУ     | ГПУ  | $S$    | ЦПУ     | ГПУ  | $S$    |
| 1 | 1.63    | 0.13 | 12.43 | 47.70   | 0.20 | 245.25 | 460.64  | 0.92 | 502.50 | 4627.61 | 8.06 | 574.39 |
| 2 | 0.14    | 0.07 | 1.87  | 5.51    | 0.17 | 33.17  | 43.58   | 0.57 | 76.00  | 436.09  | 5.22 | 83.48  |

Метод Годунова не является идеальным с точки зрения распараллеливания на ГПУ, поскольку точное решение задачи Римана предполагает большое количество ветвлений алгоритма и пересылок данных, снижающих производительность работы видеокарты, по сравнению с приближенным подходом, основанном на схеме Рое.

**5. Ударная труба.** Расчеты течения в ударной трубе проводятся в трехмерной постановке. Дискретизация уравнений Эйлера или Навье–Стокса осуществляется на неструктурированной сетке с кубическими ячейками при помощи метода конечных объемов. Для явной дискретизации по времени используются схема Эйлера или двухшаговая схема Рунге–Кутты. Расчет потоков производится при помощи схем первого (схема Годунова) или второго порядка (используется ограничитель *minmod*). Задача о распаде произвольного разрыва решается точным итерационным методом Годунова или приближенным методом Рое. При решении вязкой задачи потоки дискретизируются при помощи схем второго порядка точности.

Длина расчетной области составляет  $L = 10$  м (рис. 8). Расчеты проводятся с использованием сеток различной разрешающей способности. Шаг сетки полагается постоянным. Самая грубая сетка содержит около  $10^4$  ячеек (сетка 1), а самая подробная сетка —  $10^7$  ячеек (сетка 4). Промежуточные сетки содержат по  $10^5$  (сетка 2) и  $10^6$  (сетка 3) ячеек. Шаг по времени составляет  $1.52 \times 10^{-5}$  с, а общее время счета —  $7.63 \times 10^{-3}$  с. Число Куранта полагается равным 0.85. Расчеты проводятся на одном модуле платформы Tesla S1070 с тактовой частотой ядер 1.44 ГГц (количество ядер составляет 256), а также на одном ядре центрального процессора AMD Phenom 2 с тактовой частотой 3 ГГц.

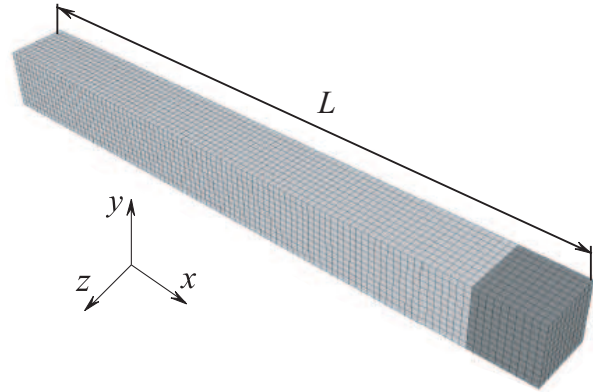


Рис. 8. Ударная труба

Время счета, а также ускорение решения задачи приводятся в табл. 3 (время, необходимое для расчета 1000 шагов, указывается в секундах). Для обозначения варианта расчета используется три индекса. Первый индекс указывает на решение невязкой (вариант 1, решаются уравнения Эйлера) или вязкой (вариант 2, решаются уравнения Навье–Стокса) задачи. Второй индекс указывает на способ дискретизации по времени (вариант А соответствует схеме Эйлера, а вариант Б — двухшаговой схеме Рунге–Кутты). Третий индекс указывает на способ дискретизации невязких потоков (варианты 1 и 2 соответствуют расчетам по схеме первого порядка с использованием точного и приближенного методов решения задачи о распаде произвольного разрыва, а варианты 3 и 4 — схемам второго порядка с использованием точного и приближенного методов решения задачи о распаде произвольного разрыва).

Таблица 3

Ударная труба. Время и ускорение вычислений

| №   | Сетка 1 |      |       | Сетка 2 |       |       | Сетка 3 |        |       | Сетка 4  |         |        |
|-----|---------|------|-------|---------|-------|-------|---------|--------|-------|----------|---------|--------|
|     | ЦПУ     | ГПУ  | S     | ЦПУ     | ГПУ   | S     | ЦПУ     | ГПУ    | S     | ЦПУ      | ГПУ     | S      |
| 1A1 | 6.63    | 0.47 | 13.96 | 69.99   | 2.64  | 26.50 | 671.22  | 22.67  | 29.61 | 6329.61  | 198.63  | 31.87  |
| 1A2 | 14.72   | 0.62 | 23.68 | 121.83  | 3.73  | 32.68 | 934.05  | 31.60  | 29.56 | 8787.27  | 207.82  | 42.28  |
| 1A3 | 12.03   | 1.29 | 9.30  | 130.33  | 10.41 | 12.52 | 1262.17 | 94.27  | 13.39 | 11018.60 | 826.18  | 13.34  |
| 1A4 | 20.66   | 1.55 | 13.33 | 193.47  | 12.81 | 15.11 | 1630.47 | 115.92 | 14.06 | 13934.40 | 872.42  | 15.97  |
| 2A1 | 18.79   | 1.36 | 13.83 | 198.96  | 10.98 | 18.12 | 1918.32 | 99.60  | 19.26 | 17485.10 | 874.99  | 19.98  |
| 2A2 | 27.79   | 1.60 | 17.35 | 261.63  | 13.31 | 19.65 | 2285.81 | 120.88 | 18.91 | 20542.40 | 914.83  | 22.46  |
| 1B1 | 12.63   | 0.91 | 13.95 | 133.88  | 5.07  | 26.39 | 1283.90 | 44.05  | 29.14 | 12163.30 | 382.90  | 31.77  |
| 1B2 | 29.14   | 1.20 | 24.30 | 242.48  | 7.29  | 33.26 | 1909.59 | 61.83  | 30.89 | 16989.20 | 406.65  | 41.78  |
| 1B3 | 23.47   | 2.50 | 9.37  | 254.6   | 20.39 | 12.49 | 1630.47 | 186.18 | 8.76  | 22484.20 | 1636.73 | 13.74  |
| 1B4 | 41.57   | 3.04 | 13.67 | 380.87  | 25.60 | 14.88 | 3227.67 | 230.86 | 13.98 | 27978.30 | 1734.85 | 16.127 |
| 2B1 | 36.83   | 2.63 | 13.98 | 388.74  | 21.25 | 18.29 | 3776.11 | 197.58 | 19.11 | 35257.70 | 1736.47 | 20.30  |
| 2B2 | 56.28   | 3.14 | 17.90 | 515.97  | 26.28 | 19.63 | 4672.10 | 240.95 | 19.39 | 40595.40 | 1822.97 | 22.27  |

При моделировании трехмерного течения в ударной трубе на подробной сетке, содержащей 10 миллионов ячеек (сетка 4), при использовании схемы Годунова (схема первого порядка) ускорение счета составляет около 42 (вариант 1A2). Решение вязкой задачи требует увеличения порядка дискретизации по пространству до второго, и ускорение падает до 22 (вариант 2A2).

Время, необходимое для расчета 1000 шагов на сетке, содержащей  $10^7$  ячеек, и используемая память приводятся в табл. 4 для различных реализаций алгоритма расчета (отношение времени счета  $S$  или количества занимаемой памяти  $M$ ). Вариант 1 соответствует расчету по схеме Годунова на одном модуле

ГПУ, вариант 2 — расчету по аналогичному алгоритму на ЦПУ, а вариант 3 — расчету с использованием коммерческого пакета Fluent (расчеты проводятся на кластере, содержащем 8 процессорных узлов).

**6. Течение в канале.** Моделирование крупных вихрей турбулентного течения вязкого сжимаемого газа в канале с квадратной формой поперечного сечения в плане проводится при  $Re_\tau = 360$  на сетке  $256 \times 64 \times 64$ . Отношение длины канала к ребру составляет  $L/H = 4$ . Сетка является равномерной по продольной координате, а в поперечном сечении узлы сетки сгущаются по направлению к стенкам канала (на стенке канала  $y^+ \sim z^+ \sim 1$ ).

Для дискретизации уравнений Навье–Стокса, замкнутых при помощи подсеточной модели Смагоринского, используется метод конечных объемов [8]. Дискретизация по времени проводится на основе трехшагового метода Рунге–Кутты. Для дискретизации конвективных потоков применяется метод кусочно-параболической реконструкции и схема Чакраварти–Ошера, а для дискретизации диффузионных потоков — центрированная разностная схема второго порядка.

Расчеты проводятся на центральном процессоре Intel Core 2 Duo с тактовой частотой 3 ГГц и графической плате GeForce GTX 480. Ускорение счета при решении задачи на разных сетках с использованием различных способов разбиения расчетной области приводится в табл. 5 (время расчета указывается в секундах для 100 шагов).

На сетке, содержащей  $256 \times 64 \times 64$  узлов, изменение способа разбиения расчетной области на блоки позволяет получить выигрыш в ускорении вычислений в 2.4 раза.

**7. Заключение.** Рассмотрены особенности параллельной реализации метода конечных объемов, предназначенного для моделирования течений вязкого сжимаемого газа на неструктурированных сетках при использовании графических процессоров общего назначения. Для программной реализации кода используется технология CUDA.

Приведено решение ряда модельных задач газовой динамики, на основе которых сопоставлены значения ускорения счета на сетках различной разрешающей способности при использовании разных способов разбиения исходных данных на блоки. Использование ГПУ позволяет ускорить расчеты в 10–50 раз (в зависимости от постановки задачи, используемых вычислительных алгоритмов и вычислительных ресурсов).

Разработанные средства численного моделирования реализованы в рамках пакета программ Логос. Пакет Логос разработан в Институте теоретической и математической физики РФЯЦ–ВНИИЭФ (Саров, Россия) и предназначен для решения широкого круга задач газовой динамики и теплообмена. Дальнейшие направления работ связаны с параллельной реализацией вычислительных алгоритмов, учитывающих турбулентность, и решением тестовых задач.

СПИСОК ЛИТЕРАТУРЫ

1. Волков К.Н., Емельянов В.Н., Карпенко А.Г., Курова И.В., Серов А.Е., Смирнов П.Г. Численное решение задач гидродинамики на графических процессорах общего назначения // Вычислительные методы и программирование. 2013. 14, № 1. 82–90.
2. Brandvik T., Pullan G. Acceleration of a 3D Euler solver using commodity graphics hardware // AIAA Paper. 2008. N 2008-607.
3. Corrigan A., Camelli F., Löhner R., Wallin J. Running unstructured grid-based CFD solvers on modern graphics hardware // AIAA Paper. 2009. N 2009-4001.
4. Corrigan A., Camelli F., Löhner R., Mut F. Semi-automatic porting of a large-scale Fortran CFD code to GPUs // Int. J. for Numerical Methods in Fluids. 2012. 69, N 2. 314–331.
5. Brandvik T., Pullan G. An accelerated 3D Navier–Stokes solver for flows in turbomachines // ASME Paper. 2009. N GT2009-60052.
6. Kampolis I.C., Trompoukis X.S., Asouti V.G., Giannakoglou K.C. CFD-based analysis and two-level aerodynamic optimization on graphics processing units // Computer Methods in Applied Mechanics and Engineering. 2010. 199,

Таблица 4

Ударная труба.  
Время и ускорение вычислений

| Вариант    | 1       | 2        | S/M   | 3       | S/M   |
|------------|---------|----------|-------|---------|-------|
| Память, Мб | 2582.28 | 2696.72  | 1.04  | 8210.16 | 3.18  |
| Время, с   | 305.29  | 14916.60 | 48.86 | 7662.00 | 25.10 |

Таблица 5

Течение в канале.  
Время и ускорение вычислений

| Сетка                     | $n_x = n_y = n_z = 4$        |       |           |
|---------------------------|------------------------------|-------|-----------|
|                           | ЦПУ                          | ГПУ   | Ускорение |
| $256 \times 64 \times 64$ | 208.09                       | 34.56 | 6.02      |
| $512 \times 64 \times 64$ | 448.47                       | 64.29 | 6.98      |
| Сетка                     | $n_x = 32, n_y = 1, n_z = 8$ |       |           |
|                           | ЦПУ                          | ГПУ   | Ускорение |
| $256 \times 64 \times 64$ | 208.09                       | 14.32 | 14.50     |

N 9–12. 712–722.

7. *Appleyard J., Drikakis D.* Higher-order CFD and interface tracking methods on highly-parallel MPI and GPU systems // *Computers and Fluids*. 2011. **46**, N 1. 101–105.
8. *Волков К.Н., Емельянов В.Н.* Моделирование крупных вихрей в расчетах турбулентных течений. М.: Физматлит, 2008.
9. *Волков К.Н.* Дискретизация уравнений Навье–Стокса на неструктурированной сетке при помощи метода контрольного объема и разностных схем повышенной разрешающей способности // *Ж. вычислит. матем. и матем. физики*. 2008. **48**, № 7. 1250–1273.
10. *Roe P.L.* Characteristic-based schemes for the Euler equations // *Annual Review of Fluid Mechanics*. 1986. **18**. 337–365.
11. *Harten A.* High resolution schemes for hyperbolic conservation laws // *J. of Computational Physics*. 1983. **49**, N 3. 357–393.
12. *Sod G.A.* A survey of several finite difference methods of systems of nonlinear hyperbolic conservation laws // *J. of Computational Physics*. 1978. **27**, N 1. 1–31.

Поступила в редакцию  
02.04.2013

---