

СИБИРСКИЕ ЭЛЕКТРОННЫЕ  
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ

Siberian Electronic Mathematical Reports

<http://semr.math.nsc.ru>

---

*Том 13, стр. 1219–1228 (2016)*

УДК 519.63

DOI 10.17377/semi.2016.13.095

MSC 35A35

ИСПОЛЬЗОВАНИЕ АДАПТИВНЫХ СЕТОК ПРИ РЕШЕНИИ  
ДВУМЕРНОГО УРАВНЕНИЯ НЕРАЗРЫВНОСТИ  
ПОЛУЛАГРАНЖЕВЫМ МЕТОДОМ

А.В. ВЯТКИН, Е.В. КУЧУНОВА, В.В. ШАЙДУРОВ

ABSTRACT. We develop a semi-Lagrangian algorithm for solving the two-dimensional advection problem. A numerical solution is constructed as a piecewise constant function on neighborhood of grid node. The proposed method is stable and gives an approximate solution with the first order of accuracy for smooth solutions. We use dynamically adaptive grid. As initial guess we consider rectangular grid.

**Keywords:** semi-Lagrangian method, adaptive grid, advection problem.

## ВВЕДЕНИЕ

В настоящее время существует большое число методов численного решения уравнения неразрывности. Первыми методами со строгим теоретическим обоснованием аппроксимации и устойчивости, были конечно-разностные схемы [1, 2]. Однако их применение накладывает жесткие ограничения на шаг по времени для обеспечения устойчивости [3, 4]. С 1960-х годов начал активно развиваться другой подход [5, 6], вытекающий из метода характеристик. Его суть состоит в том, что вдоль характеристики уравнение неразрывности можно переписать в виде обыкновенного дифференциального уравнения. При этом условие Куранта —Фридрихса —Леви выполняется автоматически без традиционных ограничений на шаг по времени. Это позволяет использовать большие шаги по времени и сократить время расчетов. В настоящее время этот подход применяется для параболических уравнений в комбинации с методами

---

VYATKIN, A.V., KUCHUNOVA, E.V., SHAYDUROV, V.V., A SEMI-LAGRANGIAN METHOD ON DYNAMICALLY ADAPTED GRID FOR TWO-DIMENSIONAL ADVECTION PROBLEM.

© 2016 Вяткин А.В., Кучунова Е.В., Шайдуров В.В.

Работа выполнена при финансовой поддержке РФФИ и Правительства Красноярского края в рамках научного проекта № 16-41-243029.

*Поступила 16 ноября 2016 г., опубликована 15 декабря 2016 г.*

конечных разностей или конечных элементов [7, 8] и поэтому часто называется полулагранжевым методом. Современные версии метода [9-11] основаны на балансовом интегральном соотношении при переходе с одного временного слоя на следующий слой. В них поиск численного решения на каждом слое по времени раскладывается на три составляющие: аппроксимация интеграла на верхнем слое по времени, на котором решение еще не известно; построение характеристик (траекторий) с верхнего временного слоя на нижний слой; приближенное вычисление интеграла на нижнем слое по времени. Для некоторых полулагранжевых алгоритмов получены результаты, позволяющие учитывать краевые условия Дирихле и обосновывать сходимость метода. Последнее десятилетие активно развиваются консервативные полулагранжевые методы, в которых численное решение удовлетворяет дискретному аналогу балансового соотношения [11, 12]. Другим важным преимуществом полулагранжевых методов является возможность использовать на разных слоях по времени разные сетки. Это преимущество привело к активному использованию адаптивных сеток [13-15]. Однако адаптивные неструктурированные сетки требуют больших вычислительных затрат при вычислении интеграла на нижнем слое по времени, что снижает их преимущество. Кроме того, выполнение дискретного аналога балансового соотношения на адаптивных сетках, как правило, приводит к необходимости использования поправочных коэффициентов [13]. Использование этих коэффициентов, с одной стороны, позволяет построить консервативные методы, а с другой стороны, вносит дополнительные погрешности в решение.

В данной работе мы предлагаем консервативный метод, который позволяет построить численное решение на адаптивных сетках без использования поправочных коэффициентов при вычислении интеграла на нижнем слое по времени. В качестве исходной сетки мы используем равномерную квадратную сетку. Критерием для последующей адаптации сетки выступает абсолютная величина дискретного аналога второй производной приближенного решения задачи. Если в некотором узле она превышает заданное значение хотя бы в одном из двух направлений, то ячейка разностной задачи делится на девять равных квадратов, а их центры берутся в качестве (дополнительных) узлов адаптированной сетки.

## 1. ПОСТАНОВКА ЗАДАЧИ

Пусть  $D = (0, 1) \times (0, 1)$  — единичный квадрат с границей  $\Gamma$ , а  $\bar{D} = D \cup \Gamma$ . На замкнутом множестве  $[0, T] \times \bar{D}$  рассмотрим двумерное уравнение адвекции

$$(1) \quad \frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{U}\rho) = 0.$$

Здесь  $\rho(t, x, y)$  — искомая функция;  $u(t, x, y)$  и  $v(t, x, y)$  известные достаточно гладкие функции вектора скорости  $\mathbf{U} = (u, v)$ . Граница  $\Gamma$  состоит из трех частей: участка втекания  $\Gamma_{\text{in}}$  на прямой  $x = 0$ , участка вытекания  $\Gamma_{\text{out}}$  на одной из сторон  $\bar{D}$  и границы с твердой стенкой  $\Gamma_{\text{rigid}}$ :  $\Gamma = \Gamma_{\text{in}} \cup \Gamma_{\text{out}} \cup \Gamma_{\text{rigid}}$ . На границе втекания  $\Gamma_{\text{in}}$  справедливо соотношение

$$(2) \quad \mathbf{U} \cdot \mathbf{n} < 0 \quad \forall (t, x, y) \in [0, T] \times \Gamma_{\text{in}},$$

где  $\mathbf{n} = (n_x(x, y), n_y(x, y))$  — вектор внешней нормали к границе  $\Gamma$ . На границе вытекания  $\Gamma_{\text{out}}$  выполняется ограничение

$$(3) \quad \mathbf{U} \cdot \mathbf{n} > 0 \quad \forall (t, x, y) \in [0, T] \times \Gamma_{\text{out}}.$$

На границе  $\Gamma_{\text{rigid}}$  выполняется условие

$$(4) \quad \mathbf{U} = (0, 0) \quad \forall (t, x, y) \in [0, T] \times \Gamma_{\text{rigid}}.$$

Искомая функция  $\rho$  удовлетворяет следующим начальным и граничным условиям:

$$(5) \quad \rho(0, x, y) = \rho_{\text{init}}(x, y) \quad \forall (x, y) \in \bar{D},$$

$$(6) \quad \rho(t, x, y) = \rho_{\text{in}}(t, x, y) \quad \forall (t, x, y) \in [0, T] \times \Gamma_{\text{in}},$$

где  $\rho_{\text{init}}(x, y)$  и  $\rho_{\text{in}}(t, x, y)$  известные функции.

## 2. ЧИСЛЕННАЯ СХЕМА

Рассмотрим в  $\bar{D}$  равномерную квадратную сетку с шагом  $h = 1/N$  и целым  $N \geq 1$ :

$$\bar{D}_h = \{(x_i, y_j) : x_i = ih, y_j = jh, i, j = 0, \dots, N\}.$$

Каждому узлу  $(x_i, y_j)$  соответствует прямоугольная окрестность

$$\Omega_{i,j} = [x_i - h/2, x_i + h/2) \times [y_j - h/2, y_j + h/2) \cap \bar{D}.$$

Для каждого узла  $(x_i, y_j)$  введем базисную функцию  $\varphi_{i,j}$ , которая равняется 1 в  $\Omega_{i,j}$  и равна 0 вне  $\Omega_{i,j}$ . Рассмотрим линейную оболочку  $L^h$  функций  $\varphi_{i,j}$ . На каждом слое по времени численное решение  $\rho^h$  задачи (1)–(6) сначала будем искать в виде кусочно-постоянной функции  $\rho^h \in L^h$ .

**2.1. Интегральные тождества.** На отрезке  $[0, T]$  введем равномерную сетку с шагом  $\tau = T/K$ ,  $K \geq 1$ , и узлами  $t_k = k\tau$ ,  $k = 0, \dots, K$ . Рассмотрим два соседних слоя по времени  $t_{k-1}, t_k$ . Пусть  $\Omega$  — произвольный четырехугольник на слое  $t_k$ . Из каждой точки  $A = (t_k, A^x, A^y)$ , лежащей на границе четырехугольника  $\Omega$ , построим траекторию движения точки  $A$  с верхнего слоя  $t_k$  на нижний слой  $t_{k-1}$ . Траектория движения определяется решением задачи Коши для системы обыкновенных дифференциальных уравнений

$$(7) \quad \begin{cases} \frac{d\tilde{x}}{dt} = u(t, \tilde{x}, \tilde{y}), \\ \frac{d\tilde{y}}{dt} = v(t, \tilde{x}, \tilde{y}) \end{cases} \quad \text{при } t \in [t_{k-1}, t_k]$$

с начальными данными

$$(8) \quad \tilde{x}(t_k) = A^x, \quad \tilde{y}(t_k) = A^y$$

Множество траекторий, выпущенных из точек границы множества  $\Omega$ , образует в плоскости  $t = t_{k-1}$ , вообще говоря, четырехугольник  $Q$  с криволинейными сторонами (рис. 1.а). Рассмотрим также плоскость втекания  $P_{\text{in}} = \Gamma_{\text{in}} \times [0, T]$ . Если  $\Omega$  расположен достаточно близко к  $P_{\text{in}}$ , то некоторые траектории её достигают и образуют на ней множество  $Q_{\text{in}}$  (рис. 1.б). В общем случае  $Q$  и  $Q_{\text{in}}$  могут быть в виде треугольника, четырехугольника, пятиугольника, прямой линии, точки или пустого множества. Положим  $d\Omega = dx dy$  и  $dS = dy dt$ . Для  $\Omega$ ,  $Q$  и  $Q_{\text{in}}$  справедливо следующее утверждение [12].

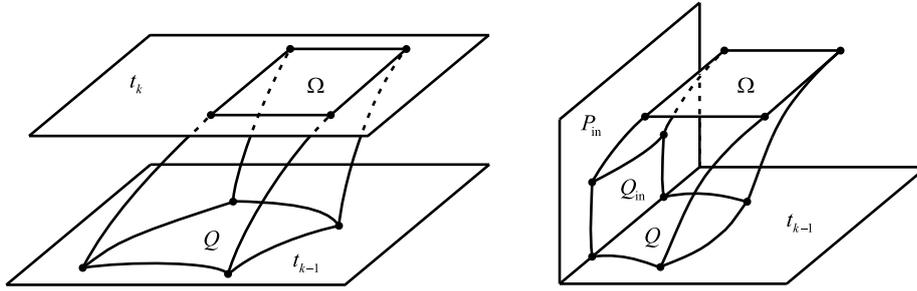


Рис.1. а. Криволинейный четырехугольник  $Q$ ; б. Выход траекторий на плоскость  $P_{\text{in}}$

**Теорема 1.** Для решения задачи (1)–(6) справедливо равенство

$$(9) \quad \int_{\Omega} \rho(t_k, x, y) d\Omega = \int_Q \rho(t_{k-1}, x, y) d\Omega + I(Q_{\text{in}}),$$

где

$$I(Q_{\text{in}}) = \begin{cases} \int_{Q_{\text{in}}} \rho_{\text{in}} u dS & \text{если } Q_{\text{in}} \neq \emptyset; \\ 0 & \text{иначе.} \end{cases}$$

**2.2. Вычисление интеграла на верхнем слое по времени.** Будем считать, что при  $t = t_{k-1}$  численное решение  $\rho^h$  нам известно на всем множестве  $\bar{D}$ . Требуется вычислить значение искомой функции  $\rho^h$  на  $k$ -том временном слое в узле  $(x_i, y_j)$ . Обозначим через  $\rho_{i,j}^{h,k}$  и  $\rho_{i,j}^k$  значения  $\rho^h(t_k, x_i, y_j)$  и  $\rho(t_k, x_i, y_j)$ , соответственно. Рассмотрим прямоугольную окрестность  $\Omega_{i,j}$  узла  $(x_i, y_j)$  в следующем виде:

$$\Omega_{i,j} = [x_i - h/2, x_i + h/2] \times [y_j - h/2, y_j + h/2] \cap \bar{D}.$$

В силу теоремы 1

$$(10) \quad \int_{\Omega_{i,j}} \rho(t_k, x, y) d\Omega = \int_{Q_{i,j}^{k-1}} \rho(t_{k-1}, x, y) d\Omega + I(Q_{\text{in},i,j}^{k-1}),$$

где

$$(11) \quad I(Q_{\text{in},i,j}^{k-1}) = \begin{cases} \int_{Q_{\text{in},i,j}^{k-1}} \rho_{\text{in}} u dS & \text{если } Q_{\text{in},i,j}^{k-1} \neq \emptyset; \\ 0 & \text{иначе.} \end{cases}$$

Для левой части соотношения (10) используем следующее приближение:

$$(12) \quad \int_{\Omega_{i,j}} \rho(t_k, x, y) d\Omega \approx \text{mes}(\Omega_{i,j}) \rho_{i,j}^k,$$

где  $\text{mes}(\Omega_{i,j})$  — площадь окрестности  $\Omega_{i,j}$ .

Чтобы вычислить правую часть (10), рассмотрим вершины  $A_n = (t_k, A_n^x, A_n^y)$ ,  $n = 1, \dots, 4$ , окрестности  $\Omega_{i,j}$ . Далее вычислим координаты вершин  $B_n =$

$(t_{k-1}, B_n^x, B_n^y)$ ,  $n = 1, \dots, 4$ , четырехугольника  $Q_{i,j}^{k-1}$ . Для этого мы решим задачу Коши (7) с начальными данными

$$(13) \quad \tilde{x}(t_k) = A_n^x, \quad \tilde{y}(t_k) = A_n^y$$

при  $t \in [t_{k-1}, t_k]$ . Численное решение задачи (7), (13) найдем методом Эйлера, а его значение на  $k - 1$ -ом слое по времени обозначим в виде  $B_n^h = (t_{k-1}, B_n^{h,x}, B_n^{h,y})$ , где

$$(14) \quad B_n^{h,x} = A_n^x - \tau u(t_k, A_n^x, A_n^y), \quad B_n^{h,y} = A_n^y - \tau v(t_k, A_n^x, A_n^y).$$

Если все четыре прямые  $(A_n^h, B_n^h)$  не пересекают плоскость втекания  $P_{in}$ , то  $I(Q_{in,i,j}^{k-1}) = 0$  (рис. 2.а). В этом случае криволинейный четырехугольник  $Q_{i,j}^{k-1}$

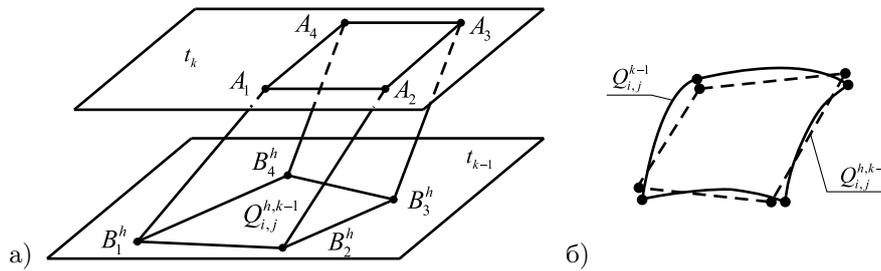


Рис. 2. а. Аппроксимация траекторий движения; б. Аппроксимация четырехугольника  $Q_{i,j}^{k-1}$

с вершинами в точках  $B_n$  мы аппроксимируем прямолинейным четырехугольником  $Q_{i,j}^{h,k-1}$  с вершинами в точках  $B_n^h$  (рис. 2.б.)

$$\int_{Q_{i,j}^{k-1}} \rho(t_{k-1}, x, y) d\Omega \approx \int_{Q_{i,j}^{h,k-1}} \rho(t_{k-1}, x, y) d\Omega.$$

В случае, если хотя бы одна прямая  $(A_n^h, B_n^h)$  пересекает  $P_{in}$ , то мы полагаем, что  $I(Q_{in,i,j}^{k-1}) \neq 0$ . Для упрощения описания алгоритма будем считать, что области  $Q_{in,i,j}^{k-1}$  и  $Q_{i,j}^{k-1}$  являются четырехугольными, как это показано на рис. 1.б. Для остальных случаев рассуждения являются аналогичными. Криволинейный четырехугольник  $Q_{in,i,j}^{k-1}$  мы аппроксимируем прямолинейным четырехугольником  $Q_{in,i,j}^{h,k-1}$ . Две вершины четырехугольника  $Q_{in,i,j}^{h,k-1}$  есть точки пересечения  $B_{in,2}^h$  и  $B_{in,3}^h$  прямых линий, аппроксимирующих соответствующие траектории движения, с плоскостью  $P_{in}$ , как это показано на рис. 3. Другие две вершины  $B_{in,1}^h, B_{in,4}^h$  определяются в виде точек пересечения плоскости  $P_{in}$  с прямыми, соединяющими точки  $B_n^h$  между собой (рис. 3). Таким образом, четыре точки  $B_{in,n}^h$ ,  $n = 1, \dots, 4$ , формируют в плоскости  $P_{in}$  четырехугольник  $Q_{in,i,j}^{h,k-1}$ , который мы используем для аппроксимации интеграла  $I(Q_{in,i,j}^{k-1})$ :

$$\int_{Q_{in,i,j}^{k-1}} \rho u(t_{k-1}, x, y) dS \approx \int_{Q_{in,i,j}^{h,k-1}} \rho u(t_{k-1}, x, y) dS.$$

Четырехугольник  $Q_{i,j}^{h,k-1}$  определяется точками  $B_{in,1}^h, B_{in,4}^h$  и  $B_2^h, B_3^h$ .

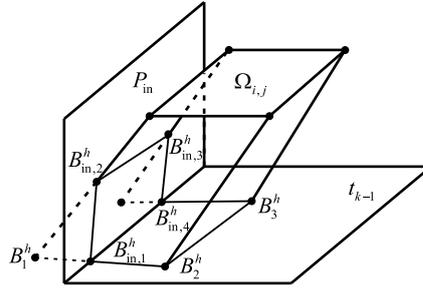


Рис. 3. Аппроксимация  $Q_{in,i,j}^{k-1}$

2.3. **Вычисление интеграла на нижнем слое по времени.** Приближим четырехугольник  $Q_{i,j}^{h,k-1}$  со сторонами  $L_n$ ,  $n = 1, \dots, 4$ , пятью прямоугольниками  $R_r$ ,  $r = 1, \dots, 5$ , стороны которых параллельны координатным осям  $Ox, Oy$ . Для этого обозначим через  $C_n = (t_{k-1}, C_n^x, C_n^y)$  центр стороны  $L_n$ ,  $n = 1, \dots, 4$ . Положим  $P_1 = (t_{k-1}, C_1^x, C_4^y)$ ,  $P_2 = (t_{k-1}, C_1^x, C_2^y)$ ,  $P_3 = (t_{k-1}, C_3^x, C_2^y)$ ,  $P_4 = (t_{k-1}, C_3^x, C_4^y)$  и сформируем четыре четырехугольника с парами вершин  $B_n^h, P_n$ . Четыре точки  $P_n$  формируют также прямоугольник  $R_5$ , интеграл по которому необходимо добавить (рис. 4.а) при аппроксимации  $I(Q_{i,j}^{h,k-1})$  или

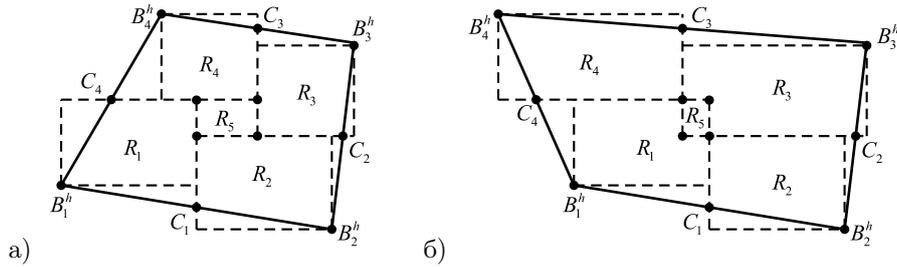


Рис. 4. а. Аппроксимация четырехугольника  $Q_{i,j}^{h,k-1}$ ; б. Наложение двух четырехугольников

вычесть из-за наложения двух прямоугольников (рис. 4.б). Несложно показать, что описанный способ построения прямоугольников  $R_n$ ,  $n = 1, \dots, 5$  приводит к тождеству

$$\text{mes}(Q_{i,j}^{h,k-1}) = \text{mes}(R_{i,j}^{k-1}),$$

где  $R_{i,j}^{k-1} = \begin{cases} \left( \bigcup_{n=1}^4 R_n \right) \cup R_5 & \text{если } R_n \text{ не имеют общих точек кроме границ;} \\ \left( \bigcup_{n=1}^4 R_n \right) & \text{иначе.} \end{cases}$

Интеграл по  $Q_{i,j}^{h,k-1}$  мы аппроксимируем интегралом по  $R_{i,j}^{k-1}$ , а затем заменим подынтегральную функцию на известное численное решение  $\rho^h(t_{k-1}, x, y)$

$$\int_{Q_{i,j}^{h,k-1}} \rho(t_{k-1}, x, y) d\Omega \approx \int_{R_{i,j}^{k-1}} \rho(t_{k-1}, x, y) d\Omega \approx \int_{R_{i,j}^{k-1}} \rho^h(t_{k-1}, x, y) d\Omega.$$

Интеграл по  $R_{i,j}^{k-1}$  представим в виде суммы интегралов

$$(15) \quad \int_{R_{i,j}^{k-1}} \rho^h(t_{k-1}, x, y) d\Omega = \sum_{n=1}^4 \int_{R_n} \rho^h(t_{k-1}, x, y) d\Omega + r_5 \int_{R_5} \rho^h(t_{k-1}, x, y) d\Omega,$$

$$\text{где } r_5 = \begin{cases} 1 & \text{если } R_n \text{ не имеют общих точек кроме границ;} \\ -1 & \text{иначе.} \end{cases}$$

Соотношение (15) позволяет вычислять интеграл от кусочно-постоянной функции по  $R_{i,j}^{k-1}$  без применения квадратурных формул.

**2.4. Учёт границы втекания.** Если окрестность  $\Omega_{i,j}$  расположена достаточно близко к плоскости  $P_{\text{in}}$ , то некоторые траектории пересекают её и образуют на ней множество  $Q_{\text{in},i,j}^{k-1}$  (рис. 1.6). В этом случае, в соответствии с (10) – (11), необходимо вычислить интеграл по области  $Q_{\text{in},i,j}^{k-1} \subset P_{\text{in}}$ . Для его вычисления мы повторим алгоритм, описанный ранее для области  $Q_{i,j}^{k-1}$ , и, в конечном счете, получим приближение

$$(16) \quad \int_{Q_{\text{in},i,j}^{k-1}} \rho_{\text{in}} u dS \approx \int_{R_{\text{in},i,j}^{k-1}} \rho_{\text{in}}^h u^h dS,$$

где  $R_{\text{in},i,j}^{k-1}$  — область построенная аналогичным образом, что и область  $R_{i,j}^{k-1}$ . Интеграл по области  $R_{\text{in},i,j}^{k-1}$  вычисляется посредством разложения множества  $R_{\text{in},i,j}^{k-1}$  на 5 прямоугольников, описанным ранее для области  $R_{i,j}^{k-1}$  способом.

**2.5. Формула для вычисления функции в узле сетки.** В соответствии с (10) – (12) численное решение  $\rho^h$  в узле  $(t_k, x_i, y_j)$  зададим по формуле

$$(17) \quad \rho_{i,j}^{h,k} = \frac{1}{\text{mes}(\Omega_{i,j})} \int_{R_{i,j}^{k-1}} \rho^h(t_{k-1}, x, y) d\Omega + \frac{1}{\text{mes}(\Omega_{i,j})} \int_{R_{\text{in},i,j}^{k-1}} \rho_{\text{in}}^h u^h dS.$$

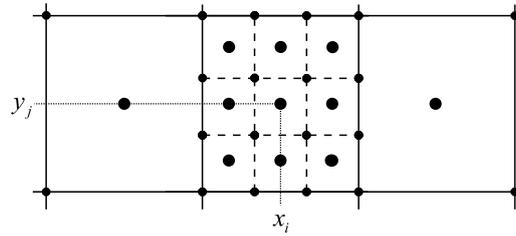
**2.6. Правила адаптации сетки.** Изначально при переходе на очередной  $k$ -ый слой по времени, мы полагаем, что сетка на  $k$ -ом слое по времени равномерная. Решение  $\rho^h$  ищется во всех узлах сетки  $\bar{D}^h$  по алгоритму описанному выше. После вычисления решения на равномерной сетке запускается алгоритм адаптации исходной сетки  $\bar{D}_h$  и поиска численного решения в новых узлах адаптивной сетки  $\bar{D}_h^{\text{ad}}$ . Под адаптацией сетки мы понимаем увеличение числа узлов сетки в тех местах вычислительной области  $\bar{D}$ , где выполняется критерий добавления новых узлов.

Будем полагать, что некоторую окрестность  $\Omega_{i,j}$  узла  $(x_i, y_j)$  необходимо разбить на окрестности меньшего размера, если справедливо хотя бы одно из следующих неравенств:

$$\frac{\rho^h(t_k, x_i + h, y_j) - 2\rho_{i,j}^{h,k} + \rho^h(t_k, x_i - h, y_j)}{h^2} > M,$$

$$\frac{\rho^h(t_k, x_i, y_j + h) - 2\rho_{i,j}^{h,k} + \rho^h(t_k, x_i, y_j - h)}{h^2} > M,$$

где  $M$  — некоторая константа (параметр адаптации). В случае дробления ячейки  $\Omega_{i,j}$  она делится на 9 равных частей (рис. 5). В геометрическом центре

Рис. 5. Дробление области  $\Omega_{i,j}$ 

каждой вновь созданной ячейки образуется узел  $(x_p^{\text{new}}, y_q^{\text{new}}) \in \bar{D}_h^{\text{ad}}$ . Численное решение задачи (1)–(6) в новых узлах сетки вычисляется описанным ранее способом.

### 3. ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Для описанного метода была проведена серия расчетов для проверки эффективности предложенного алгоритма. В качестве одного из тестов была рассмотрена задача движения круговой области  $D_c$  радиуса  $r_c = 0.1$  вокруг центра вычислительной области  $D$  по кругу радиуса  $R_c = 0.25$ . В расчетах полагалось, что в начальный момент времени центр области  $D_c$  расположен в точке  $(0.75, 0.5)$ , а полный оборот вокруг центра области  $D$  происходит за время  $T = 1$ . Для этого функции скорости были заданы в следующем виде:

$$u(t, x, y) = -2\pi\sqrt{(x-0.5)^2 + (y-0.5)^2} \sin(2\pi t),$$

$$v(t, x, y) = 2\pi\sqrt{(x-0.5)^2 + (y-0.5)^2} \cos(2\pi t).$$

В качестве точного решения задачи (1)–(6) рассмотрена функция

$$\rho_{\text{exact}}(t, x, y) = \begin{cases} 1, & \text{если } (x - x_c(t))^2 + (y - y_c(t))^2 \leq r_c^2; \\ 0, & \text{если } (x - x_c(t))^2 + (y - y_c(t))^2 > r_c^2; \end{cases}$$

где  $(x_c, y_c)$  — координаты центра области  $D_c$ , которые имеют вид

$$x_c = 0.5 + R_c \cos(2\pi t), \quad y_c = 0.5 + R_c \sin(2\pi t).$$

Начальные и граничные условия заданы в виде следующих функций:

$$\rho_{\text{init}}(x, y) = \rho_{\text{exact}}(0, x, y), \quad \rho_{\text{in}}(t, x, y) = 0.$$

При проведении расчетов в качестве начальной сетки была использована квадратная сетка с заданным параметром  $N$ . Для оценки сходимости численного решения к точному решению была использована норма пространства  $L_1$

$$\|\rho^h(T, x, y)\| = \int_D |\rho^h(T, x, y)| d\Omega.$$

Введем функцию погрешности  $\delta\rho^h(t, x, y) = \rho(t, x, y) - \rho^h(t, x, y)$ . Результаты расчетов для алгоритма без использования адаптации сетки при  $\tau = 3h$  представлены в таблице 1. Последний столбец содержит информацию о времени, которое потребовалось для проведения расчетов. Из таблицы 1 видно, что ба-

Таблица 1. Результаты расчетов без адаптации

$N \times N$	$\ \rho^h(0, x, y)\ $	$\ \rho^h(T, x, y)\ $	$\ \delta\rho^h(T, x, y)\ $	время, сек
$120 \times 120$	$3.14165581E - 02$	$3.14165581E - 02$	$1.11E - 02$	46.8
$240 \times 240$	$3.14158587E - 02$	$3.14158587E - 02$	$7.83E - 03$	396.5
$480 \times 480$	$3.14159297E - 02$	$3.14159297E - 02$	$5.57E - 03$	3150.9

балансовое соотношение выполняется для численного решения, при этом увеличение числа узлов сетки ведет к увеличению точности расчетов. Аналогичные результаты расчетов для алгоритма с использованием адаптации сетки при  $\tau = h$  представлены в таблице 2.

Таблица 2. Результаты расчетов с адаптацией сетки с одним уровнем вложенности

$N \times N$	$\ \rho^h(0, x, y)\ $	$\ \rho^h(T, x, y)\ $	$\ \delta\rho^h(T, x, y)\ $	время, сек
$40 \times 40$	$3.14165581E - 02$	$3.14165581E - 02$	$1.11E - 02$	53.5
$80 \times 80$	$3.14158587E - 02$	$3.14158587E - 02$	$7.83E - 03$	349.2
$160 \times 160$	$3.14159297E - 02$	$3.14159297E - 02$	$5.57E - 03$	2304.3

Как видно из результатов расчетов, например, для  $N = 160$  из таблицы 2, использование адаптации сетки позволяет сократить время расчетов при тех же значениях шага по времени  $\tau = 1/480$  и нормы погрешности  $\delta\rho^h$ . При этом адаптация сетки не влияет на выполнение балансового соотношения для численного решения. Для  $t = T$ ,  $N = 320$  на рис. 6.а изображено численное решение, полученное без адаптации, а на рис. 6.б представлено решение, полученное на адаптивной сетке с одним уровнем вложенности.

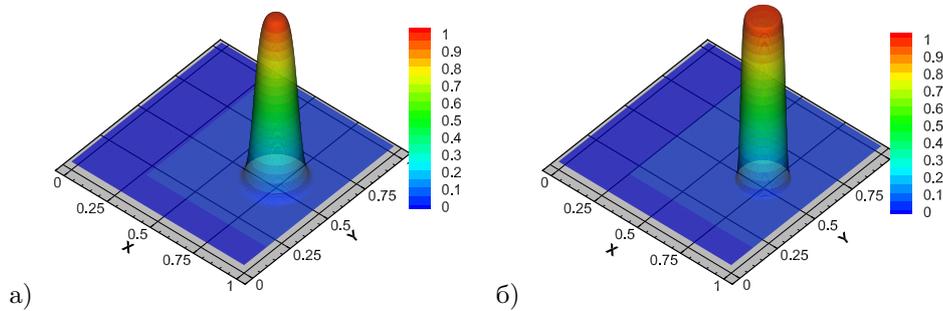


Рис. 6. а. Решение, полученное без адаптации; б. Решение, вычисленное на адаптивной сетке с одним уровнем вложенности

## ЗАКЛЮЧЕНИЕ

Для двумерного уравнения неразрывности описан численный метод, основанный на точном тождестве двух пространственных интегралов на соседних слоях по времени. Представленный подход имеет первый порядок сходимости для гладких решений. Численное решение ищется на адаптивных сетках

без использования поправочных коэффициентов при вычислении интеграла на нижнем слое по времени. При этом для численного решения выполняется балансовое соотношение. Использование адаптивных сеток позволило сократить вычислительные затраты и время расчетов. Вычислительные эксперименты полностью подтверждают теоретические выкладки.

## REFERENCES

- [1] S.K. Godunov, V.S. Ryaben'kii, *Difference Schemes*, Nauka, Moscow, 1977. Zbl 0374.65002
- [2] V. Vazov, J. Forsight, *Difference Method for Solutions of Partial Differential Equations*, Nauka, Moscow, 1963.
- [3] V.S. Ryaben'kii, A.F. Filippov, *About Atability of Difference Equations*, Gostechizdat, Moscow, 1956.
- [4] N.S. Bahvalov, N.P. Zhidkov, G.M. Kobelnikov, *Numerical Methods*, Phizmatlit, Moscow, 2003.
- [5] K.M. Magomedov, *Method of Characteristics for numerical modelling of space gas flow*, Journal of Numerical Mathematics and Mathematical Physics, **6**: 2 (1966), 313–325.
- [6] Wiin-Nielson, *On the application of trajectory methods in numerical forecasting*, Tellus, **11** (1959), 180–186.
- [7] O. Pironneau, *On the Transport-Diffusion Algorithm and Its Applications to the Navier-Stokes Equations*, Numerische Mathematik, **38** (1982), 309–332. Zbl 0505.76100
- [8] J.Jr. Douglas, T.F. Russell, *Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures*, SIAM J. Numer. Anal., **19**:5 (1982), 871–885. Zbl 0492.65051
- [9] A.V. Vyatkin, E.V. Kuchunova, *A parallel semi-Lagrangian algorithm for advection equation*, Educational Resources and Technologies, **2** (14) (2016), 423–429.
- [10] T.N. Phillips, A.J. Williams, *Conservative semi-Lagrangian finite volume schemes*, Numer. Meth. Part. Diff. Eq., **17**:4 (2001), 403–425. Zbl 0987.76063
- [11] T. Arbogast, Wen-Hao Wang, *Convergence of a fully conservative volume corrected characteristic method for transport problems*, SIAM. J. Numer. Anal., **48**:3 (2010), 797–823. Zbl 1215.35102
- [12] A. Efremov, E. Karepova, V. Shaydurov, A. Vyatkin, *A Computational Realization of a Semi-Lagrangian Method for Solving the Advection Equation*, Journal of Applied Mathematics, **2014** (2014), Article ID 610398. doi 10.1155/2014/610398
- [13] A. Iske, M. Käser, *Conservative semi-Lagrangian advection on adaptive unstructured meshes*, Numer. Meth. Part. Diff. Eq., **20**:3 (2004), 388–411. Zbl 1048.65084
- [14] F. Losasso, R. Fedkiw, S. Osher, *Spatially adaptive techniques for level set methods and incompressible flow*, Comput. Fluids, **35**:10 (2006), 995–1010. Zbl 1177.76295
- [15] K.M. Terekhov, M.A. Olshanskii, Y.V. Vassilevski, *A semi-Lagrangian method on dynamically adapted octree meshes*, Russ. J. Numer. Anal. Math. Modelling, **30**:6 (2015), 363–380. Zbl 1328.65054

ALEXANDER VLADIMIROVICH VYATKIN  
 INSTITUTE OF COMPUTATIONAL MODELLING SB RAS,  
 AKADEMGORODOK, 50/44,  
 660036, KRASNOYARSK, RUSSIA  
*E-mail address*: vyatkin@icm.krasn.ru

ELENA VLADIMIROVNA KUCHUNOVA  
 SIBERIAN FEDERAL UNIVERSITY,  
 79 SVOBODNY PR,  
 660041, KRASNOYARSK, RUSSIA  
*E-mail address*: EKuchunova@sfu-kras.ru

VLADIMIR VIKTOROVICH SHAYDUROV  
 INSTITUTE OF COMPUTATIONAL MODELLING SB RAS,  
 AKADEMGORODOK, 50/44,  
 660036, KRASNOYARSK, RUSSIA