

УДК 004.932:532.5

## Обработка изображений векторных полей с применением линейной интегральной свертки

Е. В. Ворожцов

Институт теоретической и прикладной механики им. С. А. Христиановича СО РАН, 630090, Новосибирск, Россия

Обсуждаются недостатки таких известных способов получения линий тока из векторных полей, как использование функций тока и интегрирование дифференциальных уравнений, справедливых вдоль линий тока. Описывается сравнительно новый метод визуализации линий тока – метод линейной интегральной свертки, который вытекает из заданного в цифровой форме векторного поля и существенно использует алгоритмы цифровой обработки изображений. Показано, что метод реализуется в три этапа: генерация изображения типа "белый шум" во всей области; сглаживание данного изображения вдоль линий тока на основе заданного векторного поля; улучшение полученного сглаженного цифрового изображения, для чего, в частности, предлагается использовать морфологическую операцию утончения сегментов линий тока. Дается количественная оценка точности метода линейной интегральной свертки с применением нескольких критериев. Приводятся некоторые результаты вычислительных экспериментов на двумерных векторных полях, точные линии тока которых известны. Показано, что метод линейной интегральной свертки является достаточно общим и универсальным и не требует для своей реализации какой-либо априорной информации об исследуемом явлении.

**Ключевые слова:** научная визуализация, векторные поля, линии тока, цифровые фильтры, линейная интегральная свертка, гидродинамика.

The shortcomings of such well-known techniques for obtaining the streamlines as the use of stream functions and the integration of differential equations, which are valid along the streamlines, are discussed. A comparatively new method for visualizing the streamlines – the method of linear integral convolution – is described. The method departs from the vector field given in digital form and employs substantially the algorithms of digital image processing. It is implemented in three stages: (1) generation of the image of the "white noise" type in the entire region; (2) the smoothing of the given image along the streamlines based on the given vector field; (3) improvement of the obtained smoothed digital image. To this end, it is proposed to use, in particular, the morphological operation of the thinning of the streamline segments. Quantitative estimates are given for the accuracy of the linear integral convolution method by using several criteria. Some results of computer experiments on two-dimensional vector fields, for which the exact streamlines are known, are presented. The method of linear integral convolution is shown to be very general and universal, it does not require any a priori information about the phenomenon under study for its realization.

**Key words:** scientific visualization, vector fields, streamlines, digital filters, linear integral convolution, hydrodynamics.

**Введение.** Научные исследования разнообразных динамических процессов часто связаны с необходимостью детального анализа этих процессов на основе имеющихся векторных полей. В свою очередь, этот анализ опирается на детальное изучение линий тока, получаемых из векторных полей. При этом к алгоритмам компьютерной визуализации линий тока предъявляются требования точности, скорости и высокой детальности картины линий тока [1].

Одним из известных способов построения линий тока по заданному векторному полю является численное интегрирование обыкновенных дифференциальных уравнений вдоль линии тока:

$$dx/dt = u, dy/dt = v, dz/dt = w, \quad (1)$$

где  $x, y, z$  – декартовы прямоугольные координаты точки в исследуемой пространственной области;  $t$  – параметр вдоль линии тока;  $u, v, w$  – проекции вектора скорости среды на оси координат  $Ox, Oy, Oz$ . Известен также способ построения линий тока как линий уровня функции тока. Оба эти способа построения линий тока реализованы в системе инженерных и научных расчетов MATLAB. Выбор данного пакета программ в настоящей работе обусловлен тем, что в его составе имеется пакет прикладных программ "Image processing toolbox", который включает более 100 функций, реализующих наиболее распространенные методы и алгоритмы обработки цифровых изображений [2]. В системе MATLAB имеется встроенная функция `streamline(X, Y, Z, U, V, W, startx, starty, startz)`, в которой численно интегрируются либо три уравнения (1), либо, в случае линий тока на плоскости, какие-либо два уравнения системы (1). Массивы  $X, Y, Z$  определяют координаты для  $U, V, W$ ; `startx, starty` и `startz` определяют стартовые положения линий тока. Основная трудность при использовании функции `streamline(...)` связана с выбором начальных точек линий тока. При неудачном выборе этих точек можно не заметить (пропустить) некоторые важные особенности изучаемого векторного поля, такие как области завихрения или рециркуляции, линии растекания, критические точки и т. п.

Известный способ построения линий тока как линий уровня функции тока имеет существенное ограничение: он применим только к двумерным векторным полям. Когда число пространственных переменных равно трем, функцию тока удастся построить лишь в нескольких частных случаях [3].

## 2. Метод LIC. Рассмотрим метод LIC ("line integral convolution" – линейная интегральная свертка).

2.1. *Описание метода.* Метод LIC, предложенный в работе [1], вызвал большой интерес различных исследователей, нуждающихся в картинах линий тока. Достоинства LIC-метода оказались столь значительными, что в настоящее время в мире насчитываются тысячи научных работ, в которых либо LIC-метод применяется в прикладных задачах, либо предлагаются его модификации.

В экспериментальной аэродинамике широко известен метод нанесения пленки из смеси масла и краски (или сажи) на поверхность обтекаемого тела. Когда поток газа обдувает поверхность тела, смесь масла и краски преобразуется в линии тока на поверхности. Данный способ визуализации течения наиболее эффективен при наблюдении отрывов потока газа и его повторных присоединений [4]. Однако следует отметить наличие элементов случайности:

- 1) так как указанная смесь наносится на поверхность вручную, толщина ее пленки различна на разных участках поверхности;
- 2) на наклонных участках поверхности толщина пленки меняется еще и под действием силы тяжести (пленка стекает);
- 3) на различных участках пленки локальная концентрация сажи в масле также неодинакова.

Тем не менее метод масляно-сажевой пленки позволяет получать очень информативные картины линий тока на поверхностях обтекаемых тел (см., например, работу [5]).

Метод LIC включает следующие основные этапы:

- генерация цифрового изображения типа "белый шум" во всей пространственной области;
- сглаживание данного изображения вдоль линий тока на основе заданного поля скоростей;
- улучшение полученного сглаженного изображения: контрастирование, выравнивание гистограммы яркостей и т. п.

Разработанная головная "MATLAB"-программа, реализующая LIC-метод, имеет имя `StreamLIC.m`. В этой программе метод LIC реализован для случая двух пространственных переменных: LIC-изображения линий тока строятся в сечении  $y = \text{const}$ , т. е. действующими пространственными переменными являются переменные  $x$  и  $z$ .

Для улучшения качества итогового LIC-изображения сначала осуществлялась интерполяция исходного поля скоростей  $(u, w)$  на более мелкую равномерную пространственную сетку в плоскости  $(x, z)$ . Это необходимо для того, чтобы не потерять информацию о мелких деталях векторного поля в областях сгущения исходной сетки [6].

Построение мелкой сетки выполнялось по формуле

$$h_f = \min(h_1, h_3) / f_{\text{magn}}, \quad (2)$$

где  $h_1, h_3$  – среднеарифметические значения шагов исходной неравномерной сетки вдоль осей  $x$  и  $z$ , соответственно;  $f_{magn}$  – задаваемый пользователем коэффициент увеличения детальности сетки при измельчении исходной сетки,  $f_{magn} \geq 1$ . Реальное ограничение сверху на  $f_{magn}$  диктуется лишь быстродействием используемого компьютера. При этом чем детальнее исходная пространственная сетка, тем меньшее значение  $f_{magn}$  можно брать. Заметим, что машинное время, необходимое для получения LIC-изображения, пропорционально величине  $f_{magn}^2$ .

Формула (2) показывает, что новая, более мелкая сетка состоит из квадратных ячеек, длина стороны ячейки равна  $h_f$ . Таково требование исходного LIC-метода. Для того чтобы ослабить данное требование и рассматривать ячейки мелкой сетки с неравными сторонами, требуется довольно существенная переработка программы StreamLIC.m. Однако в этом нет необходимости. Следует отметить, что в работах [7, 8] LIC-метод обобщен на случай криволинейной пространственной расчетной сетки.

Величины  $u, w$  интерполировались из центров исходных ячеек в центры ячеек более мелкой квадратной сетки с помощью встроенной "MATLAB"-функции `interp2(x, z, U, xf, zf, 'linear')`, где  $U$  – одна из величин  $u, w$ ;  $(x, z)$  – координаты точек исходной сетки, в которых заданы значения  $U$ ;  $(xf, zf)$  – координаты центров ячеек более мелкой сетки; при задании аргумента `'linear'` реализуется билинейная интерполяция, при задании аргумента `'cubic'` – бикубическая сплайн-интерполяция.

Обозначим через  $W_{ik}$  числовое значение "белого шума", генерируемого в центре ячейки мелкой сетки. "Белый шум"  $\{W_{ik}\}$  генерировался по формуле  $W_{ik} = \text{randn}(N_{3f}, N_{1f})$  (`randn` – "MATLAB"-функция, формирующая массив размера  $N_{3f} \times N_{1f}$ , элементами которого являются случайные величины, распределенные по нормальному закону с математическим ожиданием 0 и среднеквадратичным отклонением 1;  $N_{1f}, N_{3f}$  – числа ячеек мелкой сетки в направлениях осей  $x$  и  $z$  соответственно). Элемент случайности, присутствующий в алгоритме LIC-метода – единственный элемент, который является общим для LIC-метода и масло-сажевого метода из экспериментальной аэродинамики. Однако метод линейной интегральной свертки имеет существенное преимущество по сравнению с масло-сажевым методом. LIC-метод позволяет определять линии тока не только на поверхностях обтекаемых тел, но и в пространстве между телами, заполненном движущейся средой.

В дальнейших вычислениях с использованием LIC-метода осуществлялось деление длины сторон мелкой сетки на величину  $h_f$  для того чтобы получить квадратные ячейки со стороной, равной единице.

Для улучшения "белого шума" в работе [1] применялась дополнительная фильтрация нижних частот с помощью фильтра "Butterworth" четвертого порядка с частотами обрезания 128, 84, 64 и 32. Подробное описание указанного фильтра приведено в [2, 9]. В разработанной программе StreamLIC.m применение фильтра "Butterworth" считалось нецелесообразным по следующим причинам:

- численная реализация этого сглаживающего фильтра требует дополнительных расходов машинного времени;
- с уменьшением частоты обрезания "белый шум" становится более размытым, что ухудшает качество LIC-изображения (см. рис. 6 в работе [1]).

Локальное поведение векторного поля можно аппроксимировать, вычисляя локальную линию тока, начинающуюся в центре ячейки  $(i, k)$  мелкой сетки. Пусть  $P_0$  – стартовая точка линии тока,  $P_0 = (x_{i_0}, z_{i_0})$ . Адвекция точки  $P_0$  вдоль линии тока вычисляется по формулам

$$\begin{aligned} x_m &= x_{m-1} + \Delta x_{m-1}, \\ z_m &= z_{m-1} + \Delta z_{m-1}, \quad m=1, 2, \dots \end{aligned} \quad (3)$$

Алгоритм вычисления приращений координат  $\Delta x_{m-1}, \Delta z_{m-1}$  при движении вдоль линии тока со скоростью  $v = (u, w)$  сводится к рассмотрению следующих восьми случаев.

*Случай 1.* Пусть  $(i_{m-1}, k_{m-1})$  – центр ячейки, в которой находится точка  $P_{m-1}$ , пусть также

$$u_{i_{m-1}, k_{m-1}} > 0, \quad w_{i_{m-1}, k_{m-1}} > 0, \quad (w/u)_{i_{m-1}, k_{m-1}} \leq 1,$$

т. е. вектор скорости, выходящий из центра ячейки  $(i_{m-1}, k_{m-1})$ , пересекает правую сторону ячейки  $(i_{m-1}, k_{m-1})$ , (рис. 1, а). Тогда, проведя элементарное геометрическое вычисление, получаем:

$$\Delta x_{m-1} = 0,5, \quad \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1}, k_{m-1}}.$$

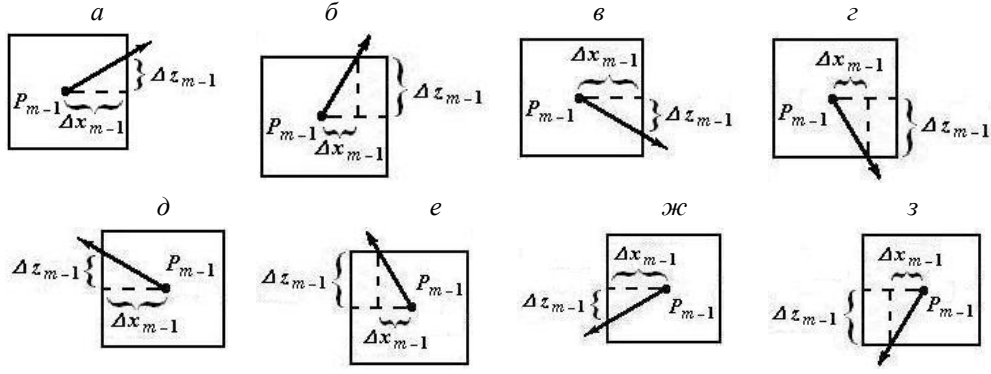


Рис. 1. Случаи расположения вектора скорости относительно сторон ячейки

Случай 2. Пусть  $u_{i_{m-1},k_{m-1}} > 0, w_{i_{m-1},k_{m-1}} > 0$ , но  $(w/u)_{i_{m-1},k_{m-1}} > 1$  (рис. 1,б). Тогда

$$\Delta z_{m-1} = 0,5, \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}}.$$

Случай 3.  $u_{i_{m-1},k_{m-1}} > 0, w_{i_{m-1},k_{m-1}} < 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| \leq 1$  (рис. 1,в). Тогда

$$\Delta x_{m-1} = 0,5, \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1},k_{m-1}} < 0.$$

Случай 4.  $u_{i_{m-1},k_{m-1}} > 0, w_{i_{m-1},k_{m-1}} < 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| > 1$  (рис. 1,г). Тогда

$$\Delta z_{m-1} = 0,5, \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}} > 0.$$

Случай 5.  $u_{i_{m-1},k_{m-1}} < 0, w_{i_{m-1},k_{m-1}} > 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| \leq 1$  (рис. 1,д). Тогда

$$\Delta x_{m-1} = 0,5, \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1},k_{m-1}} > 0.$$

Случай 6.  $u_{i_{m-1},k_{m-1}} < 0, w_{i_{m-1},k_{m-1}} < 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| > 1$  (рис. 1,е). Тогда

$$\Delta z_{m-1} = 0,5, \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}} > 0.$$

Случай 7.  $u_{i_{m-1},k_{m-1}} < 0, w_{i_{m-1},k_{m-1}} < 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| \leq 1$  (рис. 1,ж). Тогда

$$\Delta x_{m-1} = -0,5, \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1},k_{m-1}} < 0.$$

Случай 8.  $u_{i_{m-1},k_{m-1}} < 0, w_{i_{m-1},k_{m-1}} < 0$ ,  $|(w/u)_{i_{m-1},k_{m-1}}| > 1$  (рис. 1,з). Тогда

$$\Delta z_{m-1} = -0,5, \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}} < 0.$$

Сравнивая случаи 1, 3, 5, 7, нетрудно заметить, что соответствующие формулы для вычисления приращений  $\Delta x_{m-1}$  и  $\Delta z_{m-1}$  можно объединить в следующие две формулы:

$$\Delta x_{m-1} = 0,5 \operatorname{sign}(u_{i_{m-1},k_{m-1}}), \quad \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1},k_{m-1}}. \quad (4)$$

Аналогичным образом сравнивая случаи 2, 4, 6, 8, эти случаи также можно объединить в две формулы

$$\Delta z_{m-1} = 0,5 \operatorname{sign}(w_{i_{m-1},k_{m-1}}); \quad \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}}. \quad (5)$$

Вычислив приращения  $\Delta x_{m-1}$  и  $\Delta z_{m-1}$  по одной из формул (4), (5), по формулам (3) легко найти координаты  $x_m$ ,  $z_m$  следующей точки  $P_m$ . Однако при этом важно обеспечить, чтобы точка  $P_m$  вошла в соседнюю ячейку сетки [1]. Для этого в соответствии с [1] в формулах (4), (5) вместо множителя 0,5 использовался множитель  $\varepsilon_1 = 0,5(1 + \varepsilon)$

( $\varepsilon$  – малое положительное число; в приведенных ниже примерах расчетов значение  $\varepsilon = 10^{-6}$ ). Итак, формулы для вычисления  $\Delta x_{m-1}$  и  $\Delta z_{m-1}$  имеют окончательный вид

– если  $|(w/u)_{i_{m-1},k_{m-1}}| \leq 1$ , то

$$\Delta x_{m-1} = \varepsilon_1 \operatorname{sign}(u_{i_{m-1},k_{m-1}}) \quad \Delta z_{m-1} = \Delta x_{m-1} (w/u)_{i_{m-1},k_{m-1}}; \quad (6)$$

– если  $|(w/u)_{i_{m-1},k_{m-1}}| > 1$ , то

$$\Delta z_{m-1} = \varepsilon_1 \operatorname{sign}(w_{i_{m-1},k_{m-1}}) \quad \Delta x_{m-1} = \Delta z_{m-1} / (w/u)_{i_{m-1},k_{m-1}}. \quad (7)$$

Локальная линия тока продолжается из центра ячейки и в отрицательном направлении, т. е. с компонентами  $(-u, -w)$ . Это обеспечивает симметрию линии тока относительно центра ячейки. Для того чтобы найти соответствующие приращения  $\Delta x'_{m-1}$ ,  $\Delta z'_{m-1}$ , в формулы (6), (7) подставлялось значение  $-u$  вместо  $u$  и значение  $-w$  вместо  $w$ .

Теперь подсчитываются соответствующие длины дуг локальной линии тока:

$$\Delta s_{m-1} = (\Delta x_{m-1}^2 + \Delta z_{m-1}^2)^{0.5}, \quad \Delta s'_{m-1} = (\Delta x'_{m-1}{}^2 + \Delta z'_{m-1}{}^2)^{0.5}.$$

Сглаживание значения яркости изображения в ячейке  $(i, k)$  мелкой сетки осуществляется по формуле ЛИС-метода:

$$S_{ik} = \frac{\sum_{m=0}^l W_{i_m, k_m} h_m + \sum_{m=0}^{l'} W'_{i_m, k_m} h'_m}{\sum_{m=0}^l h_m + \sum_{m=0}^{l'} h'_m}, \quad (8)$$

где

$$h_m = \int_{s_{m-1}}^{s_{m-1} + \Delta s_{m-1}} k(w) dw, \quad h'_m = \int_{s'_{m-1}}^{s'_{m-1} + \Delta s'_{m-1}} k(w) dw, \quad (9)$$

$s_0 = 0$ ,  $s_m = s_{m-1} + \Delta s_{m-1}$ . Величина  $l$  в соотношении (8) равна такому значению  $i$ , при котором  $s_i \leq L \leq s_{i+1}$ , ( $L$  – длина локальной линии тока, задаваемая в терминах единичной длины стороны ячейки мелкой сетки). В работе [1] обсуждалось оптимальное значение параметра  $L$ , которое должно задаваться пользователем. С учетом рекомендаций [1] определено, что с точки зрения качества ЛИС-изображения оптимальными являются значения  $L$  в интервале  $4 < L \leq 10$ . Следует отметить, что при расчете выходного сигнала  $S_{ik}$  по формуле (8) суммарный расход времени пропорционален  $L^2$ .

Функция  $k(w)$  в (9) – это ядро свертки. В работе [1] приводились аргументы в пользу применения функции  $k(w)$  вида

$$k(w) = \frac{1 + \cos(cw)}{2} \frac{1 + \cos(dw + \beta)}{2}. \quad (10)$$

Здесь первый множитель  $(1/2)(1 + \cos(cw))$  соответствует ядру, применяемому в фильтре Hanning, обсуждение которого приведено, например, в работе [9]. Множитель  $(1/2)(1 + \cos(dw + \beta))$  – это функция ряби (ripple function);  $c$  и  $d$  – константы наращивания (dilation),  $\beta$  – сдвиг по фазе функции ряби, задаваемый в радианах.

Вследствие простоты ядра  $k(w)$  (10), интегралы в (9) легко вычисляются в аналитическом виде

$$\begin{aligned} \int_a^b k(w) dw &= \frac{1}{4} \left[ b - a + \frac{\sin(bc) - \sin(ac)}{c} + \frac{\sin(bd + \beta) - \sin(ad + \beta)}{d} \right] + \\ &+ \frac{1}{4} \left[ \frac{\sin(b(c-d) - \beta) - \sin(a(c-d) - \beta)}{2(c-d)} + \frac{\sin(b(c+d) + \beta) - \sin(a(c+d) + \beta)}{2(c+d)} \right]. \end{aligned} \quad (11)$$

Из (11) следует, что необходимо задавать различные значения  $c$  и  $d$ , чтобы избежать обращения знаменателя в нуль.

Множители  $(1/2)(1 + \cos(cw))$  и  $(1/2)(1 + \cos(dw + \beta))$  имеют периоды  $T_1 = 2\pi/c$  и  $T_2 = 2\pi/d$  соответственно, т. е. периоды зависят от констант  $c, d$ . Если в функции ряби задавать низкую частоту (т. е. малые значения  $d$ ), то с изменением сдвига по фазе  $\beta$  эта функция будет существенно меняться. Это проявляется в LIC-изображении как периодическое размывание и укрупнение (от англ. steepening – увеличение крутизны профиля) профилей яркости. Использование в функции ряби более высокой частоты приводит к оконному фильтру с почти постоянной выходной частотой, так как общая форма фильтра меняется незначительно. Но тогда отрезки линий тока будут более короткими. Если частота ряби превосходит предел Найквиста [9], определяемый размером пикселя, то длины отрезков линий тока становятся равными нулю. Как указывалось в [1], за счет увеличения пространственного разрешения (увеличения знаменателя  $f_{magn}$  в формуле (2)) всегда можно добиться и оптимальной выходной частоты, и увеличения длин сегментов линий тока, но это, как отмечалось выше, приводит к увеличению объема вычислений, необходимых для получения LIC-изображения.

Очевидно, что оптимальный подбор констант  $c, d, \beta$  для каждого исследуемого течения дает дополнительные возможности для улучшения качества линий тока, генерируемых по LIC-методу.

Полученное по формуле (8) LIC-изображение  $S = \{S_{ik}\}$  может непосредственно использоваться для визуализации линий тока. Однако имеются возможности для дальнейшего улучшения качества LIC-изображения, предложенные в работах [6-8]. В программе `StreamLIC.m` реализованы два способа улучшения LIC-изображения, предложенные в [6].

Первый способ состоит в том, чтобы использовать LIC-изображение  $S$  в качестве исходного для следующей итерации по LIC-методу. Иными словами, на второй итерации вместо исходного "белого шума"  $W$  в формуле (8) используется LIC-изображение  $S$ . В результате получаются линии тока, которые различаются в большей степени, чем при выполнении лишь одной итерации по LIC-методу. Как указывалось в [6], дальнейшие итерации (третья и т. д.) не приводят к значительным различиям в результатах по сравнению со случаем, когда LIC-алгоритм выполняется дважды. Кроме того, суммарное потребляемое машинное время прямо пропорционально числу итераций по LIC-методу. Поэтому рекомендуется выполнять не более двух итераций.

Другой способ улучшения LIC-изображения, предложенный в работе [6], заключается в приближенном выравнивании гистограммы яркости исходного "белого шума". Фильтр выравнивания гистограммы яркости сдвигает значения  $W_{ik}$  к крайним значениям диапазона. Для приближенного выравнивания гистограммы в [6] предлагалось перевычислять значения  $W_{ik}$  по формуле

$$W_{ik} = \begin{cases} W_{ik}^\alpha, & W_{ik} > 0, \\ -|W_{ik}|^\alpha, & W_{ik} \leq 0. \end{cases} \quad (12)$$

В (12) величина  $\alpha$  задается пользователем; в работе [6] для  $\alpha$  рекомендован диапазон  $5 \leq \alpha \leq 7$ .

Так как вычисления по формуле (8) – это операция сглаживания исходного LIC-изображения, то в результате происходит потеря контрастности получаемого изображения. В работе [10] предлагалось применять контрастирование LIC-изображения по формуле

$$S'_{ik} = \tilde{S}_{ik}^{\frac{4}{(\tilde{S}_{ik}+1)^5}}, \quad (13)$$

где  $\tilde{S}_{ik}$  – изображение, получаемое отображением LIC-изображения  $S_{ik}$  на интервал  $[0,1]$ . Заметим, что если  $0 \leq \tilde{S}_{ik} \leq 1$ , то показатель степени в (13) варьируется от  $1/8$  до  $4$ .

Теперь приведем некоторые результаты численных экспериментов с использованием LIC-метода для течения среды с круглым вихрем, функция тока которого имеет вид

$$\psi(x, z) = (1/2) [(x - X_c)^2 + (z - Z_c)^2] \quad (14)$$

( $X_c, Z_c$ ) – координаты центра вихря). Тогда с помощью формул  $u = \partial\psi/\partial z$ ,  $w = -\partial\psi/\partial x$  находим:  $u(x, z) = z - Z_c$ ,  $w(x, z) = -(x - X_c)$ . Вектор вихря скорости  $\text{rot } \mathbf{v} = (0, 2, 0)$  имеет лишь одну ненулевую компоненту, так что  $|\text{rot } \mathbf{v}| = 2$ .

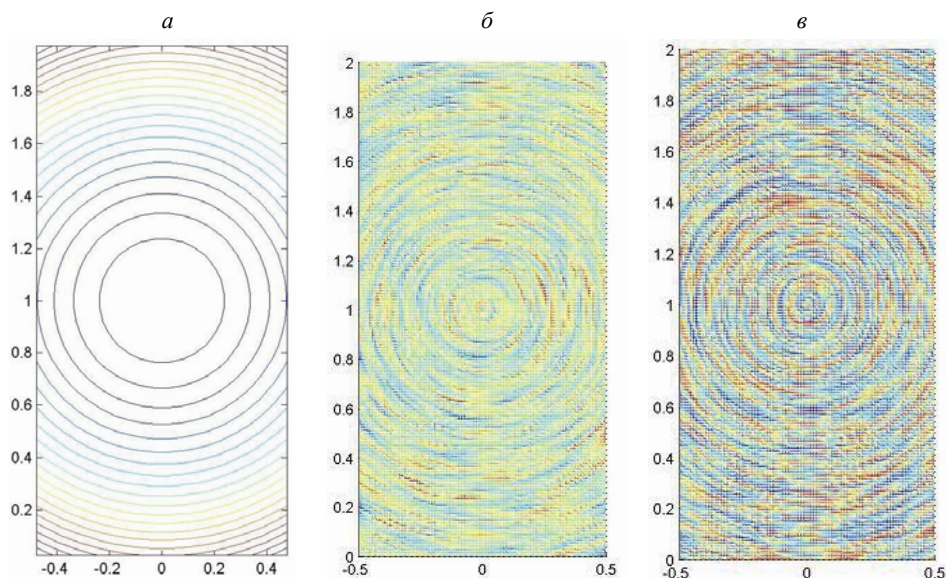


Рис. 2. Линии тока круглого вихря:

а – линии тока, полученные как линии уровня функции тока  $\psi(x,z)$ ; б – LIC-изображение  $IW$ , цвет  $C = IW$ ; в – LIC-изображение  $IW$ , цвет  $C$  вычислен по формуле (15) при  $\gamma = 0,5$

На рис. 2,а показаны линии тока рассматриваемого течения, полученные как линии уровня функции тока (14) с помощью MATLAB-функции `contour.m`. Видно, что линии тока – концентрические окружности с центром в точке  $(X_c, Z_c)$ .

Следуя работе [9], обозначим через  $IW$  результат однократного применения метода LIC к "белому шуму"  $W$ , через  $IIW = IS$  – результат двукратного применения метода LIC. В примерах расчетов по LIC-методу, приведенных на рис. 2,б, в, в соотношении (10) использовались значения  $d = 0,1$ ,  $\beta = 0,15$ . Для того чтобы выполнялось неравенство  $c \neq d$ , было выбрано значение  $c = 0,05$ . Это позволяло получить LIC-изображения, сопоставимые по качеству (прежде всего, по длинам сегментов линий тока, ухватываемым LIC-методом) с приведенными в работе [1]. На рис. 2,б показано цветное LIC-изображение  $IW$ , полученное при вызове MATLAB-функции `mesh(xx, zz, IW)`, где окраска пропорциональна высоте поверхности  $IW$ . Если в список параметров функции `mesh` ввести четвертый параметр  $C$ , управляющий локальным цветом, то можно усилить цветовое различие между соседними линиями тока на LIC-изображении. С помощью нескольких численных экспериментов эмпирически было найдено, что задание цвета по формуле

$$C_{ik} = \begin{cases} S_{ik}^\gamma, & S_{ik} > 0, \\ -|S_{ik}|^\gamma, & S_{ik} \leq 0 \end{cases} \quad (15)$$

с показателем  $\gamma = 1/\alpha$  приводит к существенному усилению цветового контраста LIC-изображения  $S = IW$  (ср. рис. 2,б и 2,в). Автором настоящей работы проводились вычислительные эксперименты, в которых не применялось приближенное выравнивание гистограммы по формуле (12), тем не менее выполнялась операция вычисления цвета согласно (15). В таких случаях целесообразность операции (15) вытекает из того, что положительную постоянную  $\gamma$  можно подобрать эмпирически из требования повышения контраста итогового LIC-изображения.

В случаях, показанных на рис. 2,б, в, в (8) использовалось значение  $L = 10$  длины окна осреднения как в положительном, так и в отрицательном направлениях вектора скорости. В формуле (2) при переходе к более мелкой сетке коэффициент увеличения  $f_{magn} = 5$ . Это означает, что на исходной сетке окно сглаживания имело сравнительно небольшой размер примерно  $4 \times 4$  ячейки.

На рис. 3,а показано LIC-изображение, полученное в результате двукратного применения LIC-метода. Сравняя рис. 3,а и рис. 2,в, видим, что повторное применение LIC приводит к улучшению связности сегментов линий тока; иными словами, средняя длина указанных сегментов увеличилась.

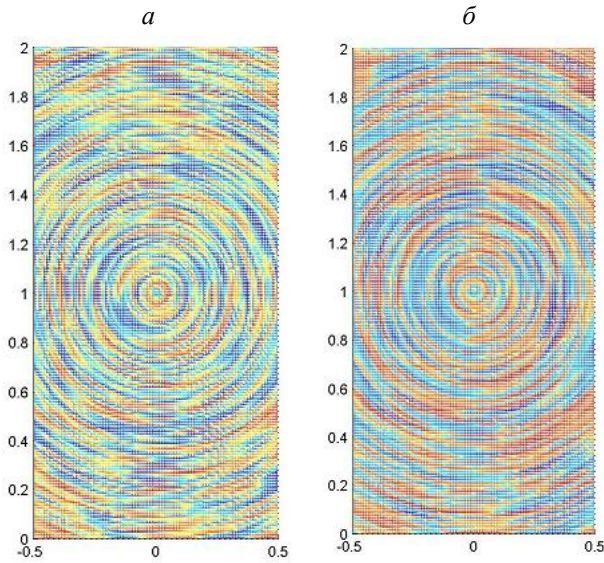


Рис. 3. Линии тока круглого вихря, полученные LIC-методом:  
 $a$  – LIC-изображение  $I^2W$ , цвет  $S$  вычислен по (15) при  $\gamma = 0,5$ ;  $b$  – LIC-изображение  $IeW$  при  $\alpha = 5$ ,  $\gamma = 0,2$

яркости изображения в ячейке  $(i, k)$  равным нулю (черный цвет), если  $S'_{ik} < th$ . Эта операция называется бинаризацией по порогу. В разработанных автором MATLAB-программах LICTest.m и StreamLIC.m наряду с генерацией цветного LIC-изображения осуществляется также бинаризация. На рис. 4 приведены результаты бинаризации двух различных LIC-изображений с различными значениями порога  $th$ .

На бинарных изображениях линии тока показаны только двумя цветами (черным и белым) вместо большого количества цветов в цветных изображениях, приведенных на рис. 2, 3. На рис. 4 видно, что толщины линий тока переменные вдоль линий тока. Этого можно избежать, применяя морфологическую операцию утончения объектов на бинарном изображении. При этом сегментами линий тока будем считать те связанные области, значения пикселей которых равны единице (белый цвет), а области изображения, значения пикселей которых равны нулю, будем

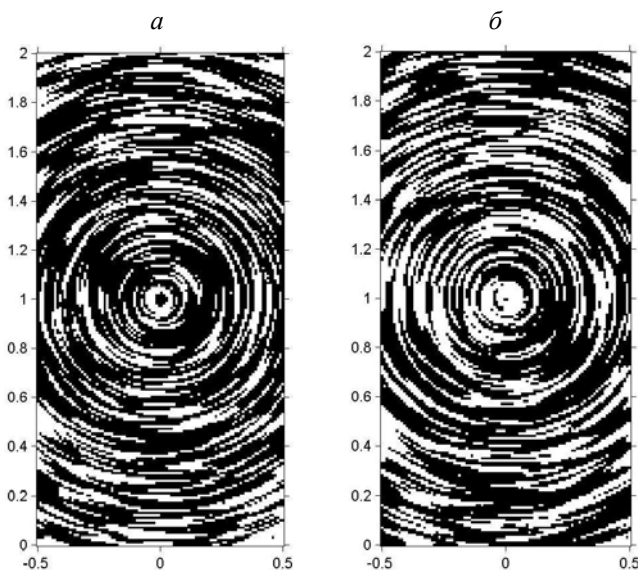


Рис. 4. Черно-белые картины линий тока, полученные бинаризацией LIC-изображений  $I^2W$ :  $th = 0,53$  ( $a$ ) и  $IeW$ ,  $th = 0,56$  ( $b$ )

Пусть, как и в работе [6],  $e$  – оператор приближенного выравнивания гистограммы яркости по формуле (12). На рис. 3,б показано LIC-изображение, полученное в результате предварительного выравнивания гистограммы "белого шума" и последующего двукратного применения метода LIC.

Поскольку во многих профильных журналах цветные рисунки не публикуются, целесообразно рассмотреть возможность инвертирования цветного LIC-изображения в черно-белую форму. Данная операция осуществлялась в два этапа. На первом этапе LIC-изображение  $S$  отображалось на интервал яркостей  $[0, 1]$  по формуле  $S' = aS + b$ , где  $a = 1,0/(S_{\max} - S_{\min})$ ,  $b = -aS_{\min}$ ,  $S_{\max} = \max_{i,k} S_{ik}$ ,  $S_{\min} = \min_{i,k} S_{ik}$ .

На втором этапе черно-белое LIC-изображение генерировалось путем вызова MATLAB-функции  $B = im2bw(S', th)$ , где  $th$  – заданное значение порога,  $0 < th < 1$ ;  $B$  – получаемое бинарное изображение. Чем выше значение порога, тем больше черных пикселей в черно-белом LIC-изображении, поскольку функция `im2bw.m` полагает значение

яркости изображения в ячейке  $(i, k)$  равным нулю (черный цвет), если  $S'_{ik} < th$ . Эта операция называется бинаризацией по порогу.

В разработанных автором MATLAB-программах LICTest.m и StreamLIC.m наряду с генерацией цветного LIC-изображения осуществляется также бинаризация. На рис. 4 приведены результаты бинаризации двух различных LIC-изображений с различными значениями порога  $th$ .

На бинарных изображениях линии тока показаны только двумя цветами (черным и белым) вместо большого количества цветов в цветных изображениях, приведенных на рис. 2, 3. На рис. 4 видно, что толщины линий тока переменные вдоль линий тока. Этого можно избежать, применяя морфологическую операцию утончения объектов на бинарном изображении. При этом сегментами линий тока будем считать те связанные области, значения пикселей которых равны единице (белый цвет), а области изображения, значения пикселей которых равны нулю, будем

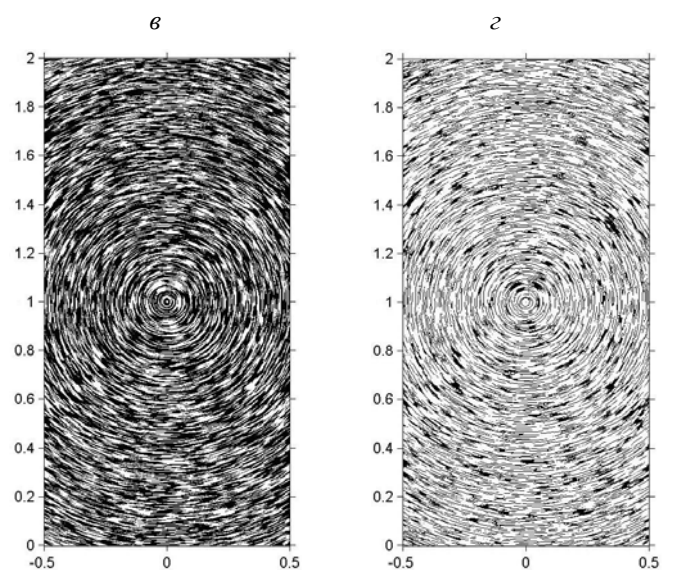


Рис. 5. Применение морфологической операции утончения:  $a$  – изображение  $bl^2W$ ;  $b$  – изображение  $itbl^2W$

считать фоном. Операция  $B_t = \text{bwmorph}(B, 'thin')$  выполняет эрозию объекта с учетом ряда условий для сохранения 8-связности участков остова [2, 11]. В результате сегменты линий тока, имевшие переменную толщину, превращаются в сегменты линий толщиной в один пиксель.

Обозначим через  $b$  оператор бинаризации изображения с помощью функции `im2bw.m`, через  $t$  – оператор морфологического утончения объектов в бинарном изображении. На изображении  $tbI^2W$  линии тока показаны белым цветом на черном фоне. Так как обычно линии тока изображаются черным цветом на белом фоне, можно осуществить операцию инвертирования изображения, при которой белые пиксели становятся черными, а черные – белыми. Обозначим этот оператор инвертирования символом  $i$ , т. е.  $itbI^2W = 1 - tbI^2W$ . На рис. 5,а показано исходное бинарное изображение  $bI^2W$ ,  $f_{magn}=20$  в (2), размер изображения  $390 \times 780$  пикселей. На рис. 5,б видно, что подавляющее большинство сегментов линий тока теперь имеет толщину в один пиксель.

На рис. 6,а показаны линии тока, полученные с помощью MATLAB-функции `streamline(...)` для задачи о безвихревом обтекании сферы потоком идеальной несжимаемой жидкости. В сечении  $y = 0$ , проходящем через центр сферы, значения составляющих скорости  $u, w$  задавались в центрах ячеек равномерной сетки в соответствии с точным решением [1]. Результат применения LIC-метода, приведенный на рис. 6,б, получен при следующих константах в (10):  $d = 0,05$ ,  $c = 2d$ ,  $\beta = 3d$ . В (8) использовалось значение  $L = 10$  длины окна осреднения как в положительном, так и в отрицательном направлениях вектора скорости.

2.2. Точность LIC-метода. В работе [10] осуществлена сравнительная количественная оценка шести методов визуализации линий тока включая метод LIC на основе векторных полей. Было сгенерировано 500 векторных полей с различным количеством критических точек различных типов (например, точек типа седла, притягивающего узла и т. д.). Реализованы следующие количественные критерии оценки точности различных алгоритмов построения линий тока.

1. Определение количества критических точек в векторном поле и вычисление на этой основе процентной доли количества векторных полей (из 500 полей), в которых конкретный метод дал неправильное количество критических точек. По данному критерию LIC-метод оказался наиболее точным из шести рассмотренных методов.

2. Определение типов выявленных критических точек. Здесь LIC-метод по точности оказался на третьем месте среди шести методов.

3. Определение средней ошибки локализации критической точки на основе изображения линий тока, построенного одним из шести методов. Положение критической точки определялось пользователем визуально на экране монитора компьютера и сразу заносилось компьютером в память, где хранились и точные координаты соответствующих критических точек. После этого вычислялась абсолютная ошибка в определении положения критической точки. В результате установлено, что метод LIC обеспечивает наиболее высокую точность определения положения критических точек по сравнению с остальными пятью методами.

4. Определение средней абсолютной угловой ошибки в направлении касательной к линии тока вблизи места ее вхождения в критическую точку. В этом случае метод LIC уступил по точности остальным пяти методам: указанная ошибка для LIC превышала  $10^\circ$ , в то время как для наилучшего по точности метода она лежала в интервале  $0,5-1,0^\circ$ .

В дополнение к описанным выше количественным критериям оценки методов визуализации линий тока на примере круглого вихря было проведено исследование влияния измельчения сетки, т. е. коэффициента  $f_{magn}$  в (2), на точность определения локальных углов наклона касательных к линиям тока в LIC-изображении  $tbI^2W$ . Пусть

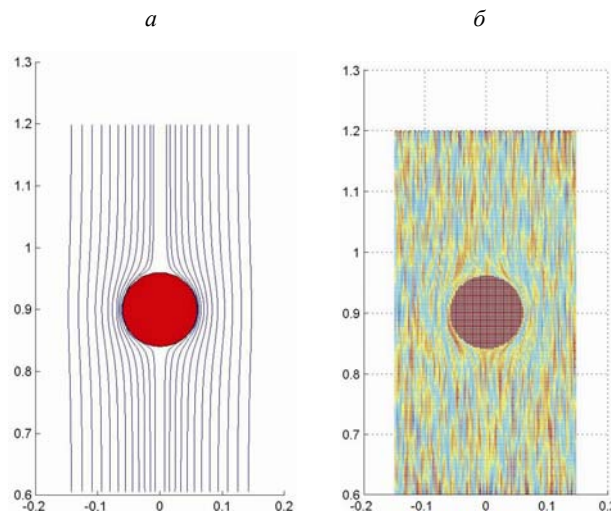


Рис. 6. Линии тока в задаче об обтекании сферы: а – линии тока, полученные с помощью функции `streamline(...)`; б – LIC-изображение  $I^2W$ , цвет  $C$  вычислен по (15) при  $\gamma = 0,5$

$tbl^2W = \{b_{k,i}\}$ ,  $k = 1, \dots, N_{3f}$ ,  $i = 1, \dots, N_{1f}$ . В пределах окна изображения размером  $3 \times 3$  с центром в пикселе  $b_{k,i}$  сегмент линии тока аппроксимируется прямолинейным отрезком, описываемым уравнением  $z = a_0 + a_1x$ . Тогда угол  $\alpha_{k,i}$  между этим отрезком и положительным направлением оси  $x$  находится из уравнения  $\text{tg } \alpha_{k,i} = (dz/dx)_{k,i} = a_1$ . В соответствии с методом наименьших квадратов составляется функция

$$F(a_0, a_1) = \sum_{m=1}^N p_m (a_0 + a_1 x_m - z_m)^2,$$

где  $(x_m, z_m)$  – координаты всех белых пикселей в пределах рассматриваемого окна изображения;  $p_m$  – заданные положительные весовые параметры (в описываемых ниже расчетах использовались значения  $p_m = 1 \forall m$ ). Из условий экстремума  $\partial F/\partial a_0 = 0$ ,  $\partial F/\partial a_1 = 0$  легко найти выражения для  $a_0$  и  $a_1$ , в частности  $a_1 = (S_0 S_3 - S_1 S_2)/(S_0 S_4 - S_1^2)$ , где

$$S_0 = \sum_{m=1}^N p_m, \quad S_1 = \sum_{m=1}^N p_m x_m, \quad S_2 = \sum_{m=1}^N p_m z_m, \quad S_3 = \sum_{m=1}^N p_m x_m z_m, \quad S_4 = \sum_{m=1}^N p_m x_m^2.$$

Вычисления по методу наименьших квадратов выполнялись только в тех окнах размером  $3 \times 3$ , в которых количество  $N$  белых пикселей удовлетворяло неравенствам  $2 < N < 6$ . Пусть  $\alpha_{k,i}^{\text{ex}}$  – точный угол наклона касательной к линии тока; в случае задачи о круглом вихре

$$\alpha_{k,i}^{\text{ex}} = \arctg[-(x_{k,i} - X_c)/(z_{k,i} - Z_c)].$$

С целью уменьшения величины  $|\alpha_{k,i} - \alpha_{k,i}^{\text{ex}}|$  выполнялись также следующие операции: если  $\alpha < 0$ , то  $\alpha = \alpha + 2\pi$ .

Далее,

если  $\alpha_{k,i} < 1,6 \wedge (4,7 < \alpha_{k,i}^{\text{ex}} < 6,29)$ , то  $\alpha_{k,i}^{\text{ex}} = 2\pi - \alpha_{k,i}$ ,

если  $4,7 < \alpha_{k,i} < 6,29 \wedge \alpha_{k,i}^{\text{ex}} < 1,6$ , то  $\alpha_{k,i} = 2\pi - \alpha_{k,i}$ ,

если  $(2,4 < \alpha_{k,i} < 2,7) \wedge (4,7 < \alpha_{k,i}^{\text{ex}} < 6,29)$ , то  $\alpha_{k,i}^{\text{ex}} = \alpha_{k,i} - \pi$ ,

если  $(2,4 < \alpha_{k,i}^{\text{ex}} < 2,7) \wedge (4,7 < \alpha_{k,i} < 6,29)$ , то  $\alpha_{k,i} = \alpha_{k,i} - \pi$ .

После этой коррекции углов  $\alpha_{k,i}$  и  $\alpha_{k,i}^{\text{ex}}$  среднеквадратичная угловая ошибка  $\Delta\alpha$  находилась по формуле

$$\Delta\alpha = \left[ (1/N_s) \sum_{m=1}^{N_s} (\alpha_{k,i} - \alpha_{k,i}^{\text{ex}})^2 \right]^{0,5}, \quad (16)$$

где  $N_s$  – количество точек линий тока в LIC-изображении  $tbl^2W$ , в которых вычислялись величины  $\alpha_{k,i}$  и  $\alpha_{k,i}^{\text{ex}}$ . На рис. 7 показана поверхность  $\Delta\alpha = \Delta\alpha(x, z)$  (углы измеряются в радианах) для случая изображения  $tbl^2W$ , инвертированный вариант которого показан на рис. 5,б. Видно, что локальные погрешности  $\Delta\alpha$  выше в следующих секторах углов  $\alpha$ :  $-\pi/4 < \alpha < \pi/4$ ,  $3\pi/4 < \alpha < 5\pi/4$ . На рис. 8 синим и красным цветом показаны линии  $\Delta\alpha = \Delta\alpha(f_{\text{magn}})$

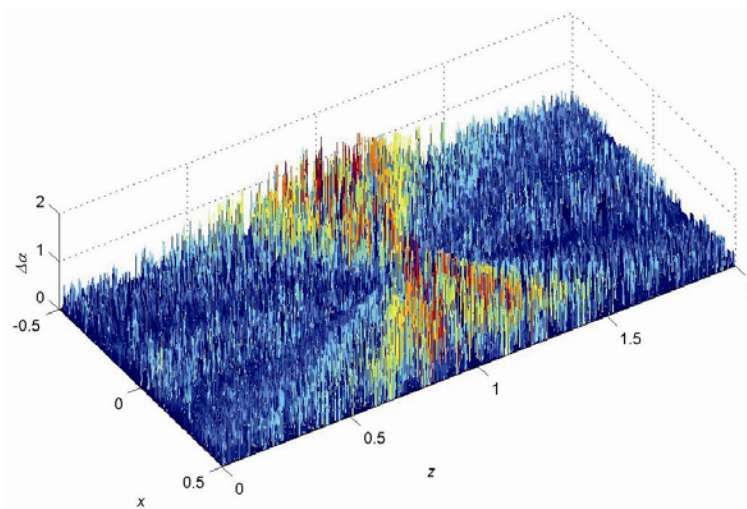


Рис. 7. Поверхность  $\Delta\alpha = \Delta\alpha(x, z)$

для случаев применения соответственно билинейной интерполяции и бикубической сплайн-интерполяции при переходе от исходной сетки к более мелкой квадратной сетке. Поскольку "белый шум"  $W$  меняется при каждом запуске MATLAB-программы генерации LIC-изображения, на рис. 8 показаны результаты первого, второго и третьего запусков программы при фиксированном значении  $f_{\text{magn}}$ .

На рис. 8 видно, что при использовании бикубической сплайн-интерполяции абсолютный максимум погрешности (16) ниже, чем в случае билинейной интерполяции. Отсюда следует предпочтительность использования бикубической сплайн-интерполяции при формировании

ЛИС-изображений. Конечно, этот метод интерполяции требует больших затрат машинного времени, чем билинейная интерполяция, но это время меньше времени, необходимого для вычислений по формулам ЛИС-метода.

Необходимо также отметить, что при увеличении  $f_{magn}$  погрешность  $\Delta\alpha$  не уменьшается.

Важно отметить следующее достоинство ЛИС-метода: поскольку вычисления по формуле (8) проводятся в каждой ячейке независимо от других, то метод ЛИС весьма удобен для распараллеливания (см. также [1]).

**Заключение.** Метод линейной интегральной свертки является универсальным методом построения линий тока по заданным локальным векторам скорости. Реализация метода проста. В отличие от известного метода интегрирования обыкновенных дифференциальных уравнений вдоль линии тока ЛИС-метод не требует задания начальных координат точек линий тока, иными словами, не требует никакой априорной информации об исследуемом течении. Поскольку в методе ЛИС сканируются все ячейки пространственной сетки, с его помощью можно успешно выявлять разнообразные локальные и нелокальные нетривиальные особенности векторного поля, некоторые из которых перечислены во введении.

Вместе с тем имеется проблема сходимости метода при измельчении сетки.

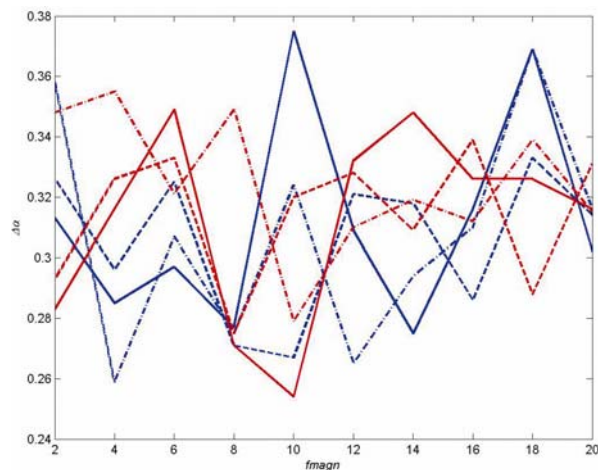


Рис. 8. Ошибка  $\Delta\alpha = \Delta\alpha(f_{magn})$ :  
синий цвет – билинейная интерполяция, красный – бикубическая сплайн-интерполяция, сплошные линии – первый запуск MATLAB-программы, штриховые – второй запуск, штрихпунктирные линии – третий запуск

#### Список литературы

1. CABRAL B., LEEDOM L. Imaging vector fields using line integral convolution // Proc. of the SIGGRAPH'93 (Special Interest Group on Computer Graphics and Interactive Techniques), Anaheim (USA). 1–6 Aug. 1993. N.Y.: ACM Press, 1993, P. 263–270.
2. РУДАКОВ П. И. Обработка сигналов и изображений. MATLAB 5.x. / П. И. Рудаков, В. И. Сафонов. М.: Диалог-МИФИ, 2000.
3. ЛОЙЦАНСКИЙ Л. Г. Механика жидкости и газа. М.: Дрофа, 2003.
4. MERZKIRCH W. Flow Visualization. N.Y.: Acad. Press, 1987.
5. ЖЕЛТОВОДОВ А. А., МАКСИМОВ А. И., ШЕВЧЕНКО А. М. Топология пространственного отрыва в условиях симметричного взаимодействия пересекающихся скачков уплотнения и волн разрежения с турбулентным пограничным слоем // Теплофизика и аэромеханика. 1998. Т. 5, № 3. С. 319–340.
6. OKADA A., LANE D. Enhanced line integral convolution with flow feature detection // Techn. Rep. NAS-96-007. NAS, June 1996.
7. FORSELL L. K. Visualizing flow over curvilinear grid surfaces using line integral convolution // Proc. of the Visualization'94. IEEE Computer Society, 17–21 Oct, 1994. Washington: D.C., 1994. P. 240–247.
8. FORSELL L. K., COHEN S. D. Using line integral convolution for flow visualization: Curvilinear grids, variable-speed animation, and unsteady flows // IEEE Transactions on Visualization and Computer Graphics. 1995. V. 1, N 2. P. 133–141.
9. STRAMPP W. Mathematische Methoden Der Signalverarbeitung // Strampp W., Vorozhtsov E. V. München, Wien: Oldenbourg Verlag, 2004.
10. LAIDLAW D. H., KIRBY R. M., DAVIDSON J. S., ET AL. Quantitative comparative evaluation of 2D vector field visualization methods // Proc. of the IEEE Visualization 2001, Oct. 24–26, 2001. IEEE Computer Society / San-Diego: Electronic Edition, 2001.
11. ГОНСАЛЕС Р. Цифровая обработка изображений // Р. Гонсалес, Р. Вудс. М.: Техносфера, 2005.

*Ворожцов Евгений Васильевич – д-р физ.-мат. наук, проф., вед. науч. сотр.  
Ин-та теорет. и прикл. механики им. С. А. Христиановича СО РАН;  
тел. (383) 330-38-04, e-mail: vorozh@itam.nsc.ru*

Дата поступления – 09.10.2009 г.