

УДК 519.6

## ВИЗУАЛИЗАЦИЯ ЛИНИЙ ТОКА И МЕТОДЫ КОМПЛЕКСНОЙ ВИЗУАЛИЗАЦИИ ДИСКРЕТНЫХ ВЕКТОРНЫХ ПОЛЕЙ

К. В. Дедкова, Д. В. Могиленских, И. В. Павлов, В. В. Федоров  
(РФЯЦ-ВНИИТФ)

Представлены алгоритмы нахождения кусочно-линейной аппроксимации линий тока дискретного векторного поля, методы визуализации и анимации линий тока, а также примеры комплексной научной визуализации векторного поля. Алгоритм нахождения линий тока базируется на подходах двойной линейной интерполяции. Алгоритмы и методы реализованы в рамках системы НВ VIZI2D.

### Введение

В работе приводится описание методов постобработки и научной визуализации (НВ) результатов численного моделирования разностными методами задач механики сплошной среды, в частности методов обработки и визуализации векторных полей. Результатом моделирования является набор состояний моделируемой системы на некоторые моменты времени, в которых система характеризуется совокупностью скалярных и векторных величин на разностной сетке. Обычно в процессе расчета накапливается довольно ограниченный набор величин как по промежуткам времени протекания процесса, так и по количеству величин — назовем это явной информацией. Постобработкой назовем обработку накопленных результатов после расчета. По явной информации можно вычислить дополнительную информацию, которую после постобработки можно представить графически, например изолинии или изоповерхности скалярных полей или линии тока (ЛТ) векторных полей.

Увеличение информативности визуализации и нахождение дополнительной информации являются важными задачами НВ. Применение алгоритмов анализа и постобработки исходной информации для получения дополнительной информации дает возможность повысить информативность НВ и получить новое качество анализа результатов. Алгоритмы аппроксимации и способы визуализации ЛТ векторных полей являются важной составляющей задачи НВ в целом. Визуализация ЛТ является одним из наиболее

информативных и эффективных способов визуального представления векторного поля, что позволяет детально передать его топологию. ЛТ — наглядная характеристика движения различных сред, столкновения, расхождения или завихренный потоков.

Данная работа посвящена алгоритмам постобработки и визуализации статических двумерных результатов моделирования. Надо отметить, что многие результаты обобщаются на трехмерный случай. При этом нет дополнительного вмешательства в методики моделирования. Разработаны разные алгоритмы, методы графической интерпретации векторных полей, которые отвечают конкретным постановкам задач [1–3].

### 1. Постановка задачи

Совокупность значений той или иной величины, заданных в каждой точке рассматриваемой области определения, называется полем этой величины. Примерами скалярных полей могут служить поле температур, поле плотностей и др. Если рассматриваемая величина — вектор, как, например, скорость, ускорение, то поле называется векторным [4].

Описание проводится на примере методов моделирования, которые используют двумерные регулярные разностные сетки. Данное предположение не ограничивает общности применимости представленных здесь алгоритмов и методов для других структур данных. Использование регулярных разностных сеток — наиболее часто

встречающийся метод дискретизации сплошной среды.

Пусть дано состояние модели на один момент времени и, не ограничивая общности, рассмотрим одну регулярную математическую область. Пусть геометрия данных, т. е. регулярная двумерная сетка, описана двумерными матрицами  $X[M, N]$ ,  $Y[M, N]$ , где  $M, N$  — размеры сетки. Пусть на сетке задана векторная величина  $U$ , т. е. в каждом узле заданы две координаты вектора  $(U_x, U_y)$  и каждая координата представляет собой скалярную величину. Векторное поле записывается в виде двух матриц  $U_x[M, N]$ ,  $U_y[M, N]$ , где  $M, N$  — размеры сетки.

Требовалось, во-первых, решить задачу кусочно-линейной аппроксимации серии ЛТ с точностью порядка пространственного разрешения разностной сетки и с достаточной скоростью вычисления аппроксимации. Вычислительная сложность данной задачи соизмерима с задачей рендеринга при формировании трехмерного изображения полигональной модели. Также было обращено внимание на возможность применения идей интерполяции освещенности внутри плоских многоугольников методами Гуро и Фонга [5], которые дают достаточное качество и хорошую скорость получения результата.

Во-вторых, на основе аппроксимированных ЛТ нужно было разработать методы комплексной НВ векторных полей.

## 2. Алгоритм аппроксимации ЛТ

Алгоритм должен основываться только на информации, которая явно дана в узлах расчетной сетки. Внутри ячеек сетки информации нет. Очевидно, что геометрически нужно опираться на ячейки расчетной сетки или на ребра ячеек. Для нахождения аппроксимации ЛТ внутри ячеек или на ребрах необходима интерполяция векторной величины из узлов. Предлагается для решения поставленной задачи НВ применить идеи алгоритмов Гуро и Фонга. Подобный подход был применен в работе [6] для цветовой интерполяции физических величин внутри ячеек. В результате алгоритм нахождения ЛТ базируется на двойной линейной интерполяции вдоль ребер ячеек и вдоль внутренних отрезков.

В контексте данной работы акцент смещается в сторону приемлемо точного и быстрого алгоритма нахождения ЛТ векторного поля для 2D-визуализации. Применение более точных ме-

тодов, например сплайнов, может быть, нужно, но это требует больших вычислительных затрат. Размерность современных решаемых задач огромна, например в механике сплошной среды, что приводит к необходимости решать очень большие системы уравнений.

Описание алгоритма проведем по шагам.

**Шаг 1: инициализация ЛТ.** Сначала происходит инициализация базовой точки (БТ) на области определения векторного поля, которая будет определять одну ЛТ. Механизм инициализации БТ может быть разный (см. разд. 3). Пусть БТ задана:  $A(X_b, Y_b)$ .

**Шаг 2: локализация БТ.** Нужно определить, какой ячейке сетки принадлежит БТ. На этом шаге работают алгоритмы локализации точки. Время локализации точки зависит от типа разностной сетки. Можно привести пример лагранжевой подвижной сетки или эйлеровой неподвижной сетки. В случае простой геометрии сетки, например равномерной прямоугольной, задача локализации БТ происходит быстро, без перебора всех ячеек. Если же сетка криволинейная, то в этом случае нужен полный перебор ячеек. Предлагается два метода.

*Метод 1 (ориентированной ячейки).* Производится последовательный полный перебор ячеек.

Рассмотрим текущую четырехугольную ячейку с индексами  $(i, j)$ ,  $(i + 1, j)$ ,  $(i + 1, j + 1)$ ,  $(i, j + 1)$  (рис. 1, а). Вычисляется габаритный прямоугольник, описывающий ячейку. Определение факта принадлежности точки прямоугольнику является тривиальным. Если БТ не принадлежит описываемому прямоугольнику, то заведомо БТ не принадлежит ячейке. Иначе нужен более точный анализ.

Можно применить следующий способ. Узлы ячеек легко ориентировать по (против) часовой стрелке и рассматривать ребра в виде векторов (рис. 1, б). Тогда для БТ можно определить ее положение относительно каждого вектора. Если для каждого вектора БТ будет справа (соответственно слева), то она принадлежит ячейке.

*Метод 2 (заметания).* В этом методе полный перебор ячеек нужен только в худшем случае и количество вычислений в среднем меньше. Рассматриваются горизонтальная ( $y = Y_b$ ) и вертикальная ( $x = X_b$ ) прямые, которые пересекаются в БТ.

Рассмотрим текущую четырехугольную ячейку с индексами  $(i, j)$ ,  $(i+1, j)$ ,  $(i+1, j+1)$ ,  $(i, j+1)$  (см. рис. 1, а). Сначала определяется факт пересечения, допустим, с горизонтальной прямой, т. е. рассматриваются координаты абсцисс узлов ячейки. Если существуют узлы с координатами  $X_1$  и  $X_2$ :  $X_1 > X_b > X_2$  или  $X_1 < X_b < X_2$ , то необходимо проверить факт пересечения с вертикальной прямой. Если нет пересечения, то переходим к следующей ячейке. Проверка факта пересечения с вертикальной прямой аналогична пересечению с горизонтальной; если ячейка не пересекает вертикальную прямую, то осуществляется переход к следующей ячейке. Если ячейка пересекает обе прямые, то ячейка анализируется более точно с помощью метода 1. Таких "подозрительных" ячеек оказывается мало.

*Примечание.* Методы 1 и 2 применимы только для выпуклых ячеек.

**Шаг 3: двойная линейная интерполяция.** Пусть известно, что БТ принадлежит ячейке с индексами  $(i, j)$ ,  $(i+1, j)$ ,  $(i+1, j+1)$ ,  $(i, j+1)$ . Необходима интерполяция значений векторной величины  $U$  в БТ. Предлагается аналогично методу Гуро ввести аналог сканирующей строки, которая проходит через БТ. Строка выбирается либо горизонтальная, либо вертикальная, что согласуется с растровой разверткой экрана. Такой выбор ориентации строк также уменьшает вычислительную нагрузку. Допустим, строка горизонтальная (на рис. 2 она обозначена буквой  $J$ ). В работе [6] рассматриваются и сравниваются разные подходы к интерполяции скалярных величин. Предлагается применить идею алгоритма Фонга двойной линейной интерполяции векторов вдоль ребер и сканирующей строки.

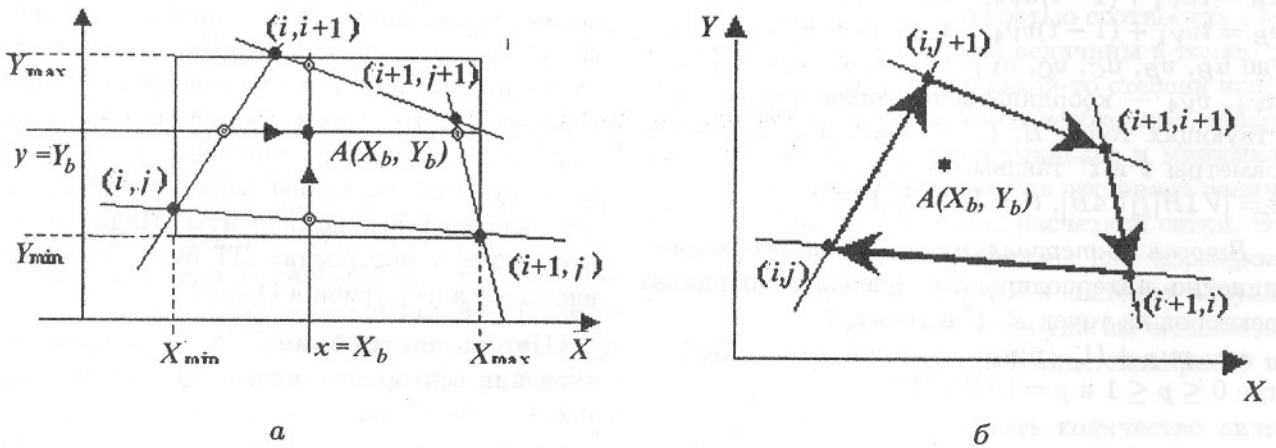


Рис. 1. Текущая четырехугольная ячейка: а — ячейка и описывающий габаритный прямоугольник; б — ячейка с ориентированными ребрами

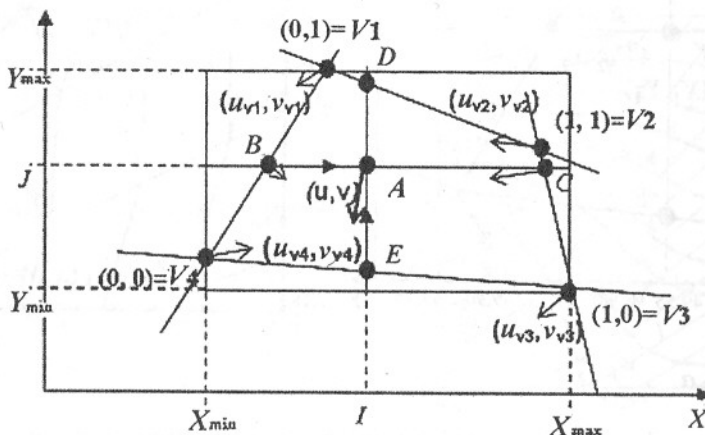


Рис. 2. Текущая ячейка с локализованной БТ А и внутренними отрезками сканирующих строк BC и DE

Не ограничивая общности, допустим, что  $i = 0$  и  $j = 0$ , т. е. индексы для текущей ячейки  $(0, 0)$ ,  $(1, 0)$ ,  $(1, 1)$ ,  $(0, 1)$ . Текущая ячейка с узловыми значениями векторной величины представлена схематично на рис. 2. Обозначим узлы так:  $V1 = (0, 1)$ ,  $V2 = (1, 1)$ ,  $V3 = (1, 0)$ ,  $V4 = (0, 0)$ . Координаты вектора  $U$  в узлах обозначим  $(u_{V4}, v_{V4})$ ,  $(u_{V1}, v_{V1})$ ,  $(u_{V2}, v_{V2})$ ,  $(u_{V3}, v_{V3})$ . Алгоритм интерполяции векторной величины сводится к двойной линейной интерполяции двух координат вектора.

**Первая интерполяция.** Для сканирующей строки вычисляются точки пересечения с ребрами ячейки. Обозначим их  $B, C$  (см. рис. 2). Теперь в эти точки необходимо интерполировать значения  $U$  из узлов ребер. Интерполирование значений векторной величины  $U$  подразумевает линейное интерполирование координат вектора  $(u, v)$ :

$$u_B = tu_{V1} + (1-t)u_{V4}; \quad u_C = tu_{V2} + (1-t^*)u_{V3};$$

$$v_B = tv_{V1} + (1-t)v_{V4}; \quad v_C = tv_{V2} + (1-t^*)v_{V3},$$

где  $u_B, v_B, u_C, v_C, u_{V1}, v_{V1}, u_{V2}, v_{V2}, u_{V3}, v_{V3}, u_{V4}, v_{V4}$  — координаты векторов для соответствующих точек  $B, C, V1, V2, V3, V4$ , а параметры  $t$  и  $t^*$  таковы:  $0 \leq t \leq 1, 0 \leq t^* \leq 1$ ;  $t = |V1B|/|V4B|, t^* = |V2C|/|V3C|$ .

**Вторая интерполяция.** Теперь необходимо линейно интерполировать значения координат векторов из точек  $B, C$  в точку  $A$ :

$$u = pu_B + (1-p)u_C; \quad v = pv_B + (1-p)v_C,$$

где  $0 \leq p \leq 1$  и  $p = |AB|/|AC|$ .

В результате двойной линейной интерполяции приближенно находится и восполняется значение векторной величины  $U_a$  в БТ (см. рис. 2). На рис. 3 представлен реальный фрагмент одной ячейки с соответствующими построениями.

Можно применить вторую двойную линейную интерполяцию, т. е. дополнительно к интерполяции по горизонтальной строке сделать и учесть интерполяцию по вертикальной строке — получить двойной порядок точности. Промежуточное значение для БТ будет найдено точнее, но, во-первых, скорость вычислений будет в 2 раза медленнее, а во-вторых, если сетка достаточно мелкая, то результат будет мало отличаться от одной билинейной интерполяции. Этот эффект был продемонстрирован в работе [6].

Можно рассмотреть возможность применения функций библиотеки OpenGL для вычисления координат векторной величины внутри ячейки по значениям в узлах, так как скорость их вычисления поддерживается на аппаратном уровне. Например, можно применить функцию сглаживания цвета или освещенности (реализация алгоритма Гуро) внутри ячейки по значениям в узлах полигона. Это означает, что компоненты векторной величины можно интерпретировать как цветовые компоненты. Надо заметить, что точность построения ЛТ будет целиком зависеть от алгоритмов в OpenGL.

**Шаг 4: построение ЛТ.** После вычисления значения векторного поля в БТ, т. е. определе-

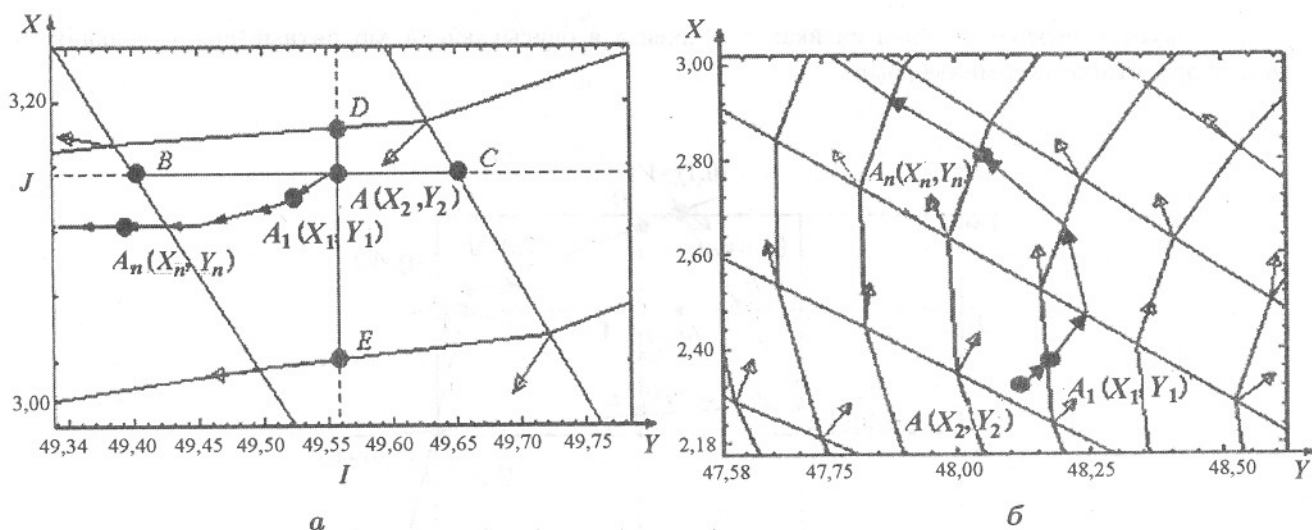


Рис. 3. Построение ЛТ: а — фрагмент текущей ячейки с локализованной внутренней точкой  $A$  и внутренними отрезками сканирующих строк  $BC$  и  $DE$  (ячеечный метод); б — пример интерполяции вдоль ребра (реберный метод)

ния направления движения ЛТ от БТ, нужно найти кусочно-линейную аппроксимацию траектории ЛТ. Предлагается два метода последовательного прослеживания или построения кусочно-линейной ЛТ.

**Метод 1 (ячеечный).** Эмпирически задается длина элементарного шага смещения от БТ вдоль вектора  $U_a$ . Например, длина шага ориентировочно определяется некоторым соотношением габаритов области определения. Зная направление шага и длину, однозначно определяем следующую точку линии тока  $A_1$  (см. рис. 3, а). Вектор  $AA_1$  назовем вектором элементарного смещения (ВЭС). Если ВЭС не пересекает ни одного ребра ячейки, то точка  $A_1$  остается в текущей ячейке, тогда нужно перейти на шаг 3 интерполяции для точки  $A_1$ . Иначе ЛТ переходит в соседнюю ячейку, и она однозначно определяется по пересеченному ребру. Меняется текущая ячейка, и осуществляется переход на шаг 3. Так последовательно происходит построение ЛТ в целом в виде кусочно-линейной кривой.

**Метод 2 (реберный).** Этот способ является более быстрым, но менее точным, он базируется на одной линейной интерполяции вдоль ребра ячейки. Из БТ выпускается вектор по направлению ВЭС до пересечения с каким-либо ребром ячейки (см. рис. 3, б). Точка пересечения  $A_1$  считается следующей точкой ЛТ, и ЛТ переходит в следующую ячейку. Далее переходим на шаг 3 интерполяции; в этом методе осуществляется только одна линейная интерполяция векторной величины вдоль пересекаемого ребра. В результате в точке  $A_1$  получаем значение векторной величины  $U_1$ . Затем из точки  $A_1$  выпускается вектор по направлению ВЭС до пересечения с каким-либо ребром данной ячейки и процесс повторяется.

Алгоритм численного нахождения кусочно-линейной аппроксимации одной ЛТ завершен.

### 3. Замечания и дополнения к алгоритму

Рассмотрим несколько подзадач, решение которых важно для визуализации ЛТ в целом.

**Остановка построения ЛТ.** Приведем примеры:

**А.** Построение ЛТ можно прекращать, когда очередной ВЭС пересечет границу расчетной области (рис. 4). Эту ситуацию можно

рассматривать более детально. Можно найти точку пересечения ВЭС с граничным ребром и вычислить с помощью одной линейной интерполяции значение векторной величины в этой точке. Тогда возможны два варианта действия. Если новый ВЭС направлен из области определения, то ЛТ прерывается на данной точке, иначе, т. е. если новый ВЭС направлен внутрь области определения, построение ЛТ продолжается.

**Б.** Данный алгоритм приближенный, поэтому в случае завихрений возможна ситуация с самопересечением (рис. 5). Поэтому необходимо постоянно проверять ЛТ на самопересечение. Проверка самопересечения — довольно дорогостоящая операция по вычислительным ресурсам, так как обычно строится серия ЛТ. Предлагаются варианты разрешения данной ситуации:

1. Длину ВЭС необходимо с обратной пропорциональностью соотносить с модулем векторной величины в точке, тогда ситуация в какой-то степени исправится. Для этого необходимо заранее вычислить максимальное и минимальное значения модуля векторной величины во всех узлах расчетной сетки. Эти значения необходимы для нормирования ВЭС. Обычно в системах визуализации эти значения уже вычислены для их отображения в виде скалярного поля.
2. Можно ограничить количество линейных частей ЛТ, а далее, если нет самопересечений, продолжить или закончить построение.

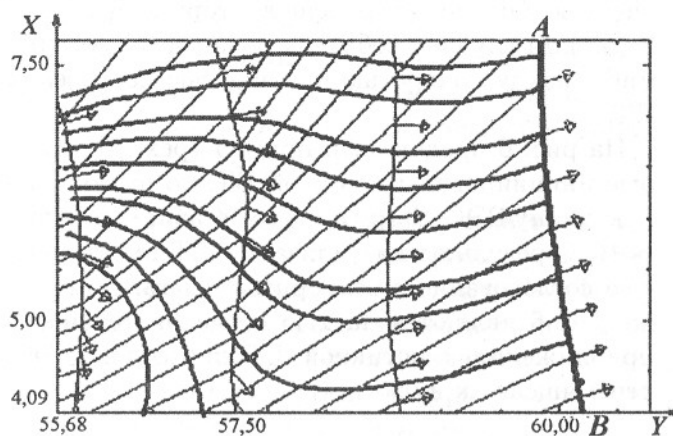


Рис. 4. Фрагмент двумерной сетки с построенными ЛТ,  $AB$  — граница сетки

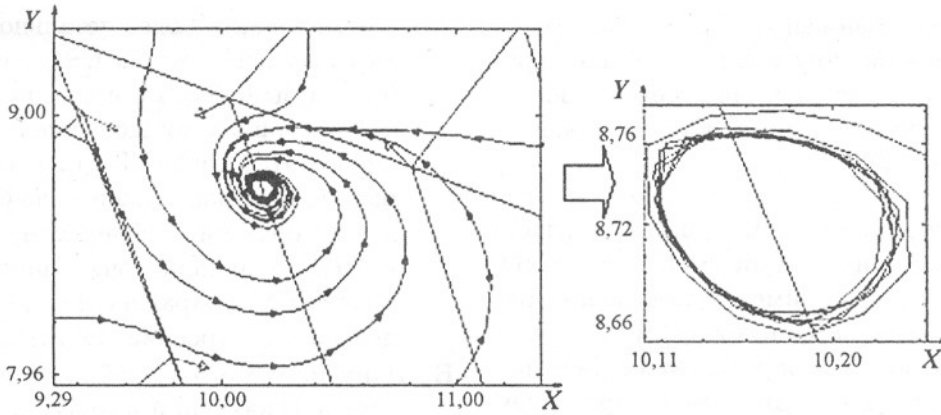


Рис. 5. Завихрения векторного поля и ЛТ. Пример самопересечения ЛТ

3. Можно обратить внимание на особенности ЛТ и явно искать самопересечения с некоторым множеством отрезков ЛТ, которое построено последним. Это связано с тем, что обычно самопересечения происходят при образовании вихрей.

**Пересечение ЛТ между собой.** Если для каждой ЛТ выбирать ВЭС разной длины, то возможна ситуация пересечения разных ЛТ между собой. Во избежание этого необходимо выбирать ВЭС одинаковой длины или выдерживать точно нормировку ВЭС в зависимости от модуля вектора в каждой текущей точке.

**Точность аппроксимации ЛТ.** Выбор длины ВЭС непосредственно влияет на точность построения ЛТ. Чем меньше длина ВЭС, тем точнее строится ЛТ, но более медленно. С другой стороны, обратим внимание на то, что внутри ячеек вообще не определена векторная функция. В данном случае необходим некоторый компромисс между точностью и скоростью построения ЛТ.

На рис. 6 представлен пример трех ЛТ, которые инициированы из одной БТ, но имеют разную длину ВЭС. Заметим, что независимо от шага ВЭС результат визуализации ЛТ имеет хорошее согласование с векторным полем. Три ЛТ на рис. 6 являются частью последовательности приближения к истинной ЛТ при длине ВЭС, стремящейся к нулю.

**Построение ЛТ в обратном направлении.** После инициализации БТ определяется соответствующий ВЭС для ЛТ векторного поля

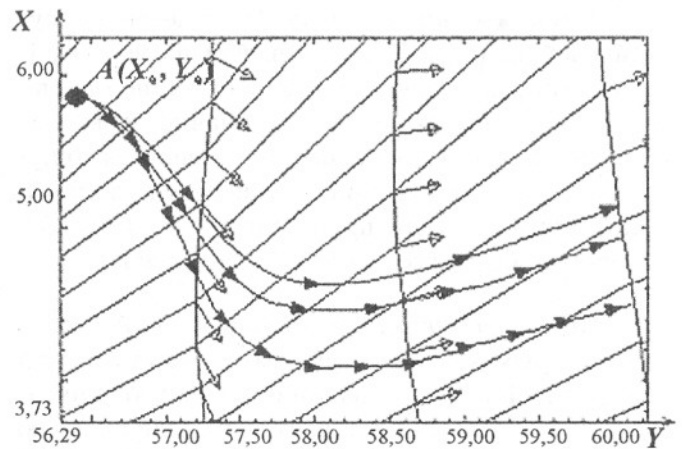


Рис. 6. Пример трех ЛТ с разной длиной ВЭС

*вперед* (т. е. все ЛТ имеют начало в своих БТ и строятся по направлению ВЭС). В этом случае ЛТ строится не полностью, так как не построена "история" ЛТ до БТ.

По описанному выше алгоритму легко продолжить ЛТ в обратном направлении от БТ. Для этого необходимо использовать вектор, обратный ВЭС. Другими словами, используется ВЭС той же длины, но противоположного направления.

**Столкновение потоков, неопределенность.** При реализации данного алгоритма построения ЛТ имеет место эффект *пилл* при столкновении потоков векторного поля (рис. 7, а). Для устранения этого эффекта применяется анализ угла между двумя последними ВЭС. Если угол близок к нулю, то построение ЛТ прекращается (рис. 7, б).

**Ячеечные векторы.** Для реализации алгоритма построения ЛТ необходимы значения  $U$

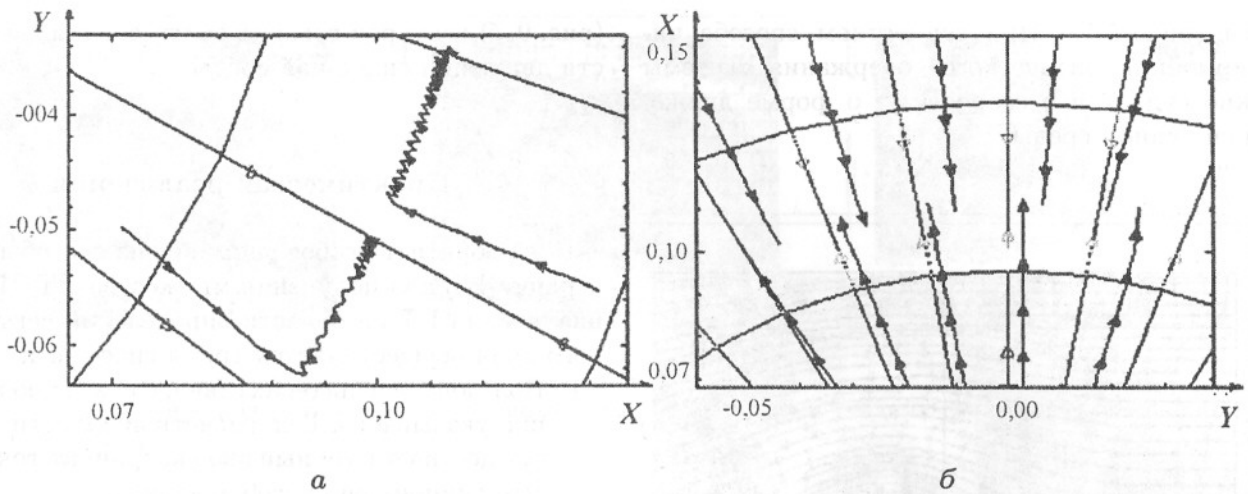


Рис. 7. Столкновение потоков векторного поля: а — эффект пила; б — эффект устранен

в узлах сетки. Векторные величины в задачах могут быть ячеечными или узловыми. Если величина узловая, то не требуется предобработки; если она ячейная, то необходимо интерполировать ее значения в узлы. Здесь тоже применяется идея линейной интерполяции Гуро — это интерполяция векторов из смежных ячеек по данному узлу в узел.

#### 4. Анимация псевродинамики ЛТ векторного поля

Рассматривается статическое состояние системы на некоторый момент времени моделирования, поэтому о настоящей динамике говорить невозможно. По этой причине представляемую динамику будем называть псевродинамикой. Несмотря на статичность, в каждом узле сетки задана вполне определенная векторная величина, у которой есть потенциальная величина изменения. Данной информации достаточно для реализации разных способов повышения информативности визуализации, что позволит более точно и качественно оценить состояние модели. Анимационные функции визуализации трудно представить статическими картинками. Представим несколько способов визуализации псевродинамики, т. е. визуализации изменения модуля векторной величины вдоль ЛТ:

1. Изменение скорости визуализации траектории ЛТ с помощью применения нормированной длины ВЭС.
2. Показ движения маркера или множества маркеров с соответствующей скоростью вдоль траектории ЛТ.

3. Прорисовка траектории ЛТ штрихами, которые периодически смещаются по направлению ЛТ.
4. Статическая визуализация с помощью цветной визуализации траектории ЛТ с использованием цветовой палитры, которая соответствует диапазону изменения модуля вектора вдоль траектории ЛТ.
5. Визуализация ЛТ векторных полей для последовательных временных разрезов. БТ в каждом разрезе можно привязать к узлам сетки, что дает возможность более адекватно инициализировать ЛТ для разных разрезов.

#### 5. Комплексная визуализация векторных полей

Комплексность НВ заключается в том, чтобы с помощью одновременного применения разных функций визуализации получить более информативную картину моделируемой системы. Приведем несколько примеров комплексной визуализации с применением ЛТ.

**Пример 1.** Методы и примеры цветовой интерпретации физического содержания моделей представлены в разных работах, например [7–9]. Существует возможность совместного применения цветового изображения физических скалярных характеристик (температуры, плотности, модуля скорости и т. д.) и методов визуализации векторных величин. Хорошим способом является одновременная визуализация ЛТ на фоне цветовой визуализации модуля векторной вели-

чины (рис. 8)\*. При совместном способе интерпретации физического содержания системы можно судить и о скорости, и о форме движения сплошной среды.

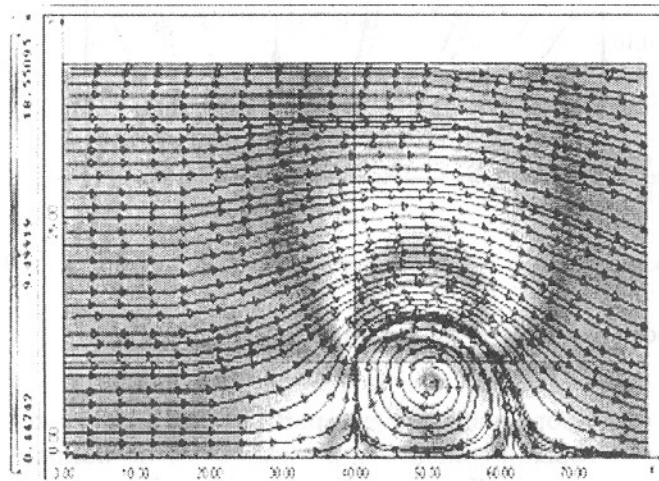


Рис. 8. Цветовая (см. сноску) визуализация модуля вектора скорости и ЛТ векторного поля

**Пример 2.** Визуализация ЛТ на фоне явной визуализации векторного поля в виде векторов является хорошим методом комплексной НВ (рис. 9). Изменение цвета (рис. 9, а) и длины

(рис. 9, б) векторов говорит об изменении скорости движения сплошной среды.

## 6. Практическая реализация

Надо обратить особое внимание на способы генерации БТ для получения множества ЛТ. Инициализация БТ на области определения векторного поля осуществляется тремя способами:

1. Пользователь интерактивно, с помощью мыши, указывает БТ на расчетной области или осуществляет точный ввод координат точки. Это одиночный способ задания ЛТ.
2. БТ генерируются процедурой случайного выбора, т. е. случайным образом некоторое множество БТ "набрасывается" на область определения векторного поля. Далее из каждой БТ строится ЛТ.
3. БТ ассоциируются с узлами расчетной сетки. Этот способ является более регулярным и его можно применять для реальной динамической визуализации векторных полей с помощью ЛТ по последовательности состояний системы на разные моменты времени.

На рис. 10 — фрагмент динамической визуализации векторного поля множеством ЛТ на разных стадиях развития процесса.

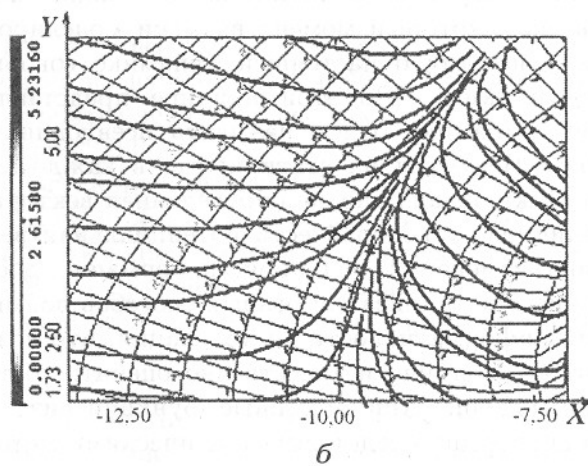
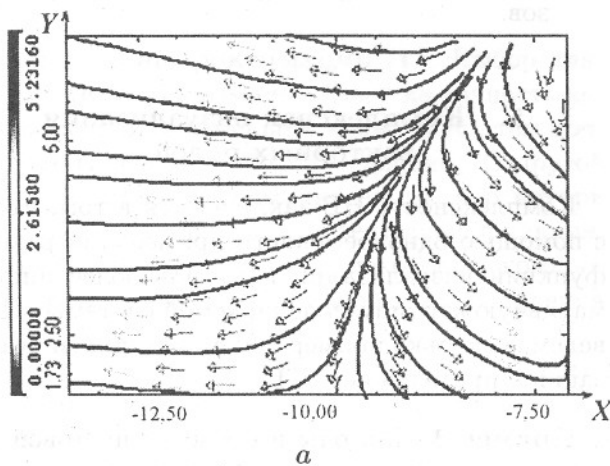


Рис. 9. Визуализация ЛТ и явная визуализация векторов цветом (см. сноску) в зависимости от их модуля: а — без учета их длин; б — с учетом их длин

\*К сожалению, черно-белое исполнение журнала не позволяет передать цвета. В данном случае об изменении цвета можно судить по интенсивности серого тона.



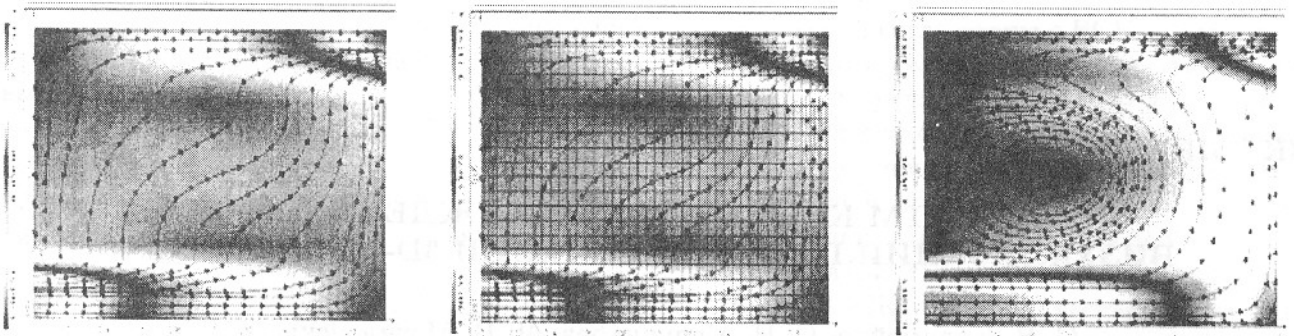


Рис. 10. Векторное поле, представленное множеством ЛТ, на разных стадиях развития процесса течения сплошной среды

### Заключение

Изложенные в работе алгоритмы и методы визуализации дискретных векторных полей реализованы в системе НВ результатов двумерного численного моделирования и экспериментов VIZI2D [7, 9].

Надо отметить, что статическая визуализация — мгновенная "фотография" состояния модели, и ЛТ показывает потенциальную тенденцию движения вещества. Методы комплексной визуализации дают возможность повышения информативности НВ, что в свою очередь является важной задачей НВ. Улучшение точности в сочетании с хорошими скоростными характеристиками остается важной задачей.

В следующих работах будут представлены обобщение на трехмерные результаты численного моделирования и динамическая визуализация в условиях постобработки.

### Список литературы

1. Сельвачев А. Ю., Аксенов А. А., Клименко С. В. Общий подход к многочастичным анимационным методам визуализации векторного поля // Тр. 6-й межд. конф. по компьютерной графике и анимации "Аниграф'98". Москва, 20–23 мая 1998 г. С. 98–102.
2. Буланов С. В., Ечкина Е. Ю., Иновенков И. Н., Леоненко А. В. Анимация перерезывания магнитных силовых линий в двумерных конфигурациях // Тр. 8-й межд. конф. по компьютерной графике и визуализации "Графикон'98". Москва, 7–11 сентября 1998 г. С. 57–58.
3. Ечкина Е. Ю., Иновенков И. Н., Павлова О. А. Визуализация двумерных и трехмерных векторных полей // Там же. С. 59–60.
4. Седов Л. И. Механика сплошной среды. Т. 1. М.: Наука, 1976.
5. Foley J. D., van Dam A., Feiner S. K., Hughes J. F. Computer Graphics: Principles and Practice. Second Edition in C. Addison-Wesley, 1997. P. 736–739.
6. Mogilenskikh D. V., Pavlov I. V. Color interpolation algorithms in visualizing results of numerical simulations // Visualization and Imaging in Transport Phenomena. Annals of the New York Academy of sciences. 2002. Vol. 972. P. 43–52.
7. Могиленских Д. В., Федоров В. В., Павлов И. В. Визуализация двумерных результатов численного моделирования физических процессов. Система визуализации "VIZI" в ОС Windows // V Забавихинские науч. чтения. Межд. конф. Снежинск, 21–25 сентября 1998 г.
8. Mogilenskikh, D. V. Nonlinear color interpretation of physical processes // Proc. Int. Conf. on Computer Graphics "Graphicon'2000". Moscow, August 28 — September 2, 2000. P. 201–211.
9. Могиленских Д. В., Павлов И. В., Федоров В. В., Мельникова С. Н., Сапожникова Е. Э. Принципы построения и функциональное содержание системы визуализации для анализа скалярных и векторных полей, заданных на двумерных регулярных сетках: Препринт № 172. Снежинск: РФЯЦ-ВНИИТФ, 2000.