УДК 519.5 Вестник СПбГУ. Сер. 10. 2014. Вып. 3

Л. М. Романовский

РЕАЛИЗАЦИЯ АЛГОРИТМА ЛОКАЛЬНОГО УКРУПНЕНИЯ ТРИАНГУЛЯЦИИ

Санкт-Петербургский государственный университет, Российская Федерация
199034, Санкт-Петербург, Университетская наб., 7/9

В работе представлена правильная триангуляция, топология которой позволяет проводить локальное укрупнение с сохранением правильности на границе и в области укрупнения. Разработан алгоритм, с помощью которого можно построить триангуляцию с локальным укрупнением по заданной исходной триангуляции с сохранением топологии исходной триангуляции в области укрупнения. Показывается изоморфизм топологий исходной и укрупненной триангуляций. Реализована программа, дающая возможность проводить многократные адаптивные рекуррентные укрупнения триангуляции, а также выполняющая построение курантовской аппроксимации исходных значений на результирующей триангуляции. Приводятся результаты тестирования алгоритма. Библиогр. 3 назв. Ил. 2. Табл. 1.

Ключевые слова: курантовская аппроксимация, аппроксимация поверхности, триангуляция, локальное укрупнение, 3D-графика, таблица инциденций.

L. M. Romanovskii

IMPLEMENTATION OF LOCAL TRIANGULATION ENLARGEMENT ALGORITHM

St. Petersburg State University, 7/9, Universitetskaya embankment, St. Petersburg,
199034, Russian Federation

In this paper we consider a two-dimensional simplex partition plane (triangulation) which authorizes a local enlargement of the conservation correctness at the border and in the consolidation area. To set the triangulation incidence table is used, which enumerates all triangles involved in triangulation. Amalgamation is proposed to conduct the union of two triangles having a common edge. Equivalent conversion of incidence table is that lines, corresponding to excluded triangles are excluded from the incidence table, while lines of triangles corresponding to joinable triangles are added to incidence table. The algorithm of enlargement, preserving the topology of the original triangulation, which allows multiple recurrent enlargement, is proposed. The article deals with the rectangular grid with integer coordinates, but presented algorithm allows curved triangulations. The author has made a computer program that implements the algorithm under consideration. On the basis of the input data the triangulation with necessary topology is formed and then locally recurrent enlargement is executed as long as the resulting error on the enlarged grid nodes approximation does not exceed the limit value. For the convenience of testing the ability to be used as input graphic image data in a common format is implemented. Presentation of results shows that the algorithm avoids consideration of grid nodes to 70 percent while maintaining a given accuracy of the approximation of the original data. Bibliogr. 3. Il. 2. Table 1.

Keywords: Kourant approximation, surface approximation, triangulation, local enlargement, 3D-graphic, incidence table.

1. Введение. Существующие алгоритмы позволяют строить аппроксимации одномерных функций на неравномерной сетке узлов, что дает возможность учитывать

Романовский Леонид Михайлович – аспирант; e-mail: Leonid.Romanovskii@gmail.com
Romanovskii Leonid Mikhailovich – post-graduate student; Leonid.Romanovskii@gmail.com

локальные особенности входных данных для последующего сжатия с сохранением допустимой погрешности.

Однако в случае обработки двумерных данных (в частности, графических изображений, карт высот, 3D-поверхностей), нельзя учитывать локальные особенности данных по нескольким измерениям одновременно. Вместо этого обычно предлагается растягивать входной многомерный поток данных в одномерный, а затем строить по нему аппроксимацию. Такой подход применяется, например, в алгоритме сжатия изображений jpeg.

В настоящей работе рассматривается алгоритм, учитывающий локальные особенности входного потока данных по двум измерениям единовременно. Реализована компьютерная программа, позволяющая проводить адаптивное локальное укрупнение триангуляции, а также выполняющая аппроксимацию исходных данных с использованием укрупненной триангуляции (см. п. 3).

2. Основные обозначения. В плоскости $\mathbb{R}^2$ рассматривается декартова система координат (x, y), точкам этой плоскости сопоставляются радиус-векторы $r = (x, y)$. Треугольник, возможно, криволинейный, с вершинами a, b, c обозначается $\|a \quad b \quad c\|$; порядок следования вершин здесь роли не играет. Если описываются несколько треугольников, например треугольник с вершинами a, b, c, треугольник с вершинами d, e, f и треугольник с вершинами u, v, w, то строим таблицу

$$\left\| \begin{matrix} a & b & c \\ d & e & f \\ u & v & w \end{matrix} \right\|,$$

называемую таблицей инциденций. Порядок следования строк в данном обозначении роли не играет.

3. Триангуляция и ее первое укрупнение. В плоскости $\mathbb{R}^2$ рассмотрим правильную (возможно, криволинейную) триангуляцию $\mathbb{T}_0$, которая характеризуется таблицей инциденций

$$\left\| \begin{matrix} r_{2i,2j} & r_{2i,2j+2} & r_{2i+1,2j+1} \\ r_{2i,2j} & r_{2i+2,2j} & r_{2i+1,2j+1} \\ r_{2i+2,2j+2} & r_{2i+2,2j} & r_{2i+1,2j+1} \\ r_{2i+2,2j+2} & r_{2i,2j+2} & r_{2i+1,2j+1} \end{matrix} \right\| \qquad \forall (i, j) \in \mathbb{Z}^2.$$

Всякую прямолинейную триангуляцию, нульмерный остов которой содержится в множестве вида $\{(ih, jh) \mid (i, j) \in \mathbb{Z}^2\}$ (здесь h – некоторое фиксированное число), называем стандартной триангуляцией.

Укрупнение триангуляции будем проводить объединением двух треугольников исходной триангуляции, имеющих общее ребро. Для этого заменим перечисленные выше треугольники на треугольник, получающийся их объединением. Эквивалентное преобразование таблицы инциденций заключается в том, что из нее исключаются строки, соответствующие объединяемым треугольникам, и добавляются строки, отвечающие результатам такого объединения, укрупненным треугольникам. Еще раз подчеркнем, что порядок строк в таблице инциденций не играет никакой роли, потому удаление и добавление строк можно осуществлять в любых местах таблицы инциденций. Таким образом, для задания укрупнения достаточно перечислить выбрасываемые и добавляемые строки. Преобразования будем задавать указанием двух строкаменяемых треугольников (в левой от стрелки части формулы); например, отображение

$$\left\| \begin{matrix} x_0 & y_0 & z_0 \\ x_1 & y_1 & z_1 \end{matrix} \right\| \to \|x_2 \quad y_2 \quad z_2\|$$

означает, что треугольники $\|x_0 \quad y_0 \quad z_0\|$ и $\|x_1 \quad y_1 \quad z_1\|$ заменяются треугольником $\|x_2 \quad y_2 \quad z_2\|$.

Рассмотрим следующие отображения:

$$\varepsilon_{ij}^l : \quad \left\| \begin{matrix} r_{2i,2j} & r_{2i+2,2j} & r_{2i+1,2j+1} \\ r_{2i+2,2j} & r_{2i+2,2j+2} & r_{2i+1,2j+1} \\ r_{2i+2,2j+2} & r_{2i,2j+2} & r_{2i+1,2j+1} \\ r_{2i,2j+2} & r_{2i,2j} & r_{2i+1,2j+1} \end{matrix} \right\| \to \left\| \begin{matrix} r_{2i,2j} & r_{2i+2,2j} & r_{2i,2j+2} \\ r_{2i+2,2j+2} & r_{2i+2,2j} & r_{2i,2j+2} \end{matrix} \right\|$$

$$\forall (i,j) \in \mathbb{Z}^2 \quad (i+j) \bmod 2 \equiv 0,$$

и

$$\varepsilon_{ij}^r : \quad \left\| \begin{matrix} r_{2i+2,2j} & r_{2i+2,2j+2} & r_{2i+1,2j+1} \\ r_{2i+2,2j+2} & r_{2i,2j+2} & r_{2i+1,2j+1} \\ r_{2i,2j+2} & r_{2i,2j} & r_{2i+1,2j+1} \\ r_{2i,2j} & r_{2i+2,2j} & r_{2i+1,2j+1} \end{matrix} \right\| \to \left\| \begin{matrix} r_{2i,2j} & r_{2i+2,2j+2} & r_{2i+2,2j} \\ r_{2i,2j} & r_{2i+2,2j+2} & r_{2i,2j+2} \end{matrix} \right\|$$

$$\forall (i,j) \in \mathbb{Z}^2 \quad (i+j+1) \bmod 2 \equiv 0.$$

Полученная с их помощью триангуляция $\mathcal{T}_1$ будет иметь вид

$$\left\| \begin{matrix} r_{2i,2j} & r_{2i+2,2j} & r_{2i,2j+2} \\ r_{2i+2,2j+2} & r_{2i+2,2j} & r_{2i,2j+2} \end{matrix} \right\|, \quad (i+j) \equiv 0 \bmod 2 \quad \forall (i,j) \in \mathbb{Z}^2,$$

$$\left\| \begin{matrix} r_{2i,2j} & r_{2i+2,2j+2} & r_{2i+2,2j} \\ r_{2i,2j} & r_{2i+2,2j+2} & r_{2i,2j+2} \end{matrix} \right\|, \quad (i+j+1) \equiv 0 \bmod 2 \quad \forall (i,j) \in \mathbb{Z}^2.$$

Утверждение. *Триангуляция $\mathcal{T}_1$ топологически изоморфна исходной триангуляции $\mathcal{T}_0$.*

Следствие. *Для стандартных триангуляций переход от триангуляции $\mathcal{T}_0$ к триангуляции $\mathcal{T}_1$ может быть осуществлен поворотом плоскости на угол $\frac{\pi}{4}$ и масштабированием ее с коэффициентом $\sqrt{2}$ относительно начала координат.*

4. Алгоритм укрупнения в данной области. Рассмотрим прямоугольную область

$$D = \{(x,y), \quad |x| \leqslant M, \quad |y| \leqslant N\},$$

где M и N – натуральные числа. Будем считать, что числа M и N достаточно велики. В области D построим целочисленную сетку узлов. Положим

$$r_{ij} = (i,\, j),$$
$$i \in \{-M, -M+1, \ \dots \ , M-1, M\},$$
$$j \in \{-N, -N+1, \ \dots \ , N-1, N\}.$$

Таким образом получаем прямоугольную сетку размером $(2M+1) \times (2N+1)$.

Определим множество D_0 следующим образом:

$$D_0 = \{(x, y) \mid \quad \varphi(x, y) \leqslant 0,\ (x, y) \in D\}$$

для некоторой измеримой функции $\varphi(x, y)$, заданной на D.

Пусть

$$T(i,j) = \left\| \begin{array}{lll} (2i, 2j) & (2i+2, 2j) & (2i+1, 2j+1) \\ (2i+2, 2j) & (2i+2, 2j+2) & (2i+1, 2j+1) \\ (2i+2, 2j+2) & (2i, 2j+2) & (2i+1, 2j+1) \\ (2i, 2j+2) & (2i, 2j) & (2i+1, 2j+1) \end{array} \right\|,$$

$$Tl(i,j) = \left\| \begin{array}{lll} (2i, 2j) & (2i+2, 2j) & (2i, 2j+2) \\ (2i+2, 2j+2) & (2i+2, 2j) & (2i, 2j+2) \end{array} \right\|,$$

$$Tr(i,j) = \left\| \begin{array}{lll} (2i, 2j) & (2i+2, 2j+2) & (2i+2, 2j) \\ (2i, 2j) & (2i+2, 2j+2) & (2i, 2j+2) \end{array} \right\|.$$

В случае, если все элементы матрицы $T(i, j)$ содержатся в области D_0, заменяем в таблице инциденций строки, описанные в матрице $T(i, j)$, на строки из матрицы $T_l(i, j)$ или $T_r(i, j)$. Выбор между $T_l(i, j)$ и $T_r(i, j)$ делается таким образом, чтобы структура укрупненной триангуляции совпадала со структурой исходной триангуляции.

Для удобства на рис. 1, *а, б* покажем, как будут выглядеть треугольники, объединенные с помощью отображений ε^l и ε^r (выделены жирными линиями)

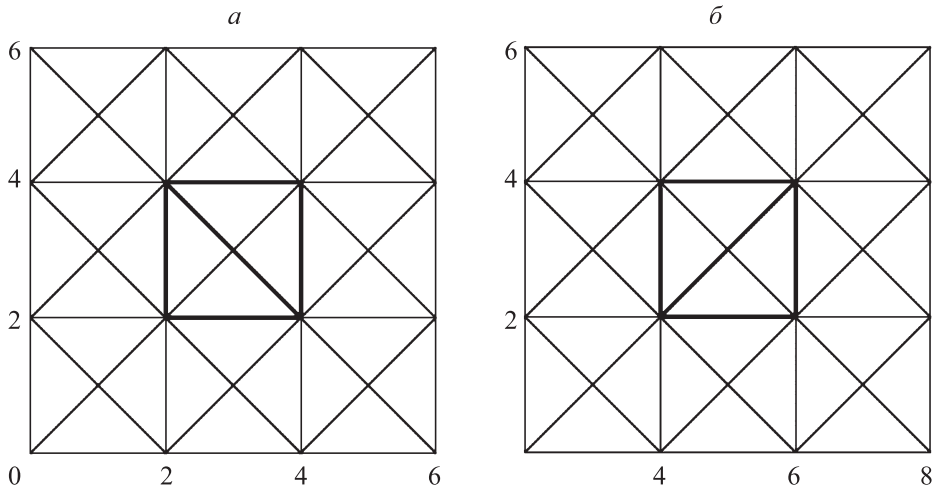

Рис. 1. Результат работы отображений ε^l (*а*) и ε^l (*б*)

Результатом работы алгоритма является триангуляция области D с укрупнением в области D_0. Полученная триангуляция описывается таблицей инциденций T_{rez}.

Приведем псевдокод рассматриваемого алгоритма:

```
for i = (−M + 1) to (M − 1) do
    for j = (−N + 1) to (N − 1) do
        if All items of T(i, j) are inside of D_0 then
            if (i + j) mod 2 = 0 then
                Add Tl(i, j) to T_res
            end
            else
                Add Tr(i, j) to T_res
            end
        end
        else
            Add T(i, j) to table T_res
        end
    end
end
```

Входными данными этой программы являются числа M и N, а также множество значений аппроксимируемой функции на сетке узлов.

Также приводится графическое изображение примера локально укрупненной триангуляции (рис. 2). Жирными линиями выделены укрупненные треугольники.

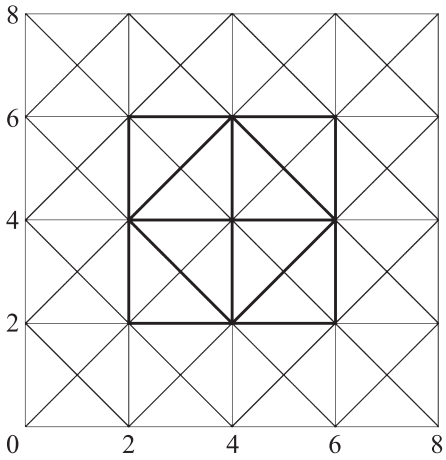

Рис. 2. Триангуляция с локальным укрупнением

5. Построение аппроксимации. Для простоты в настоящей работе будет рассматриваться только курантовская аппроксимация [1, 2]. Вместе с тем при решении практических задач возможно применение любых других методов аппроксимации двумерных потоков данных, в частности вэйвлетных. Построение таких приближений рассмотрено в [3].

Реализация проведена автором в виде компьютерной программы. Язык программирования – Java, проект осуществлен в Eclipse IDE. Для графического отображения триангуляции применяется библиотека Swing.

6. Входные и выходные данные. Входными параметрами алгоритма являются триангуляция в виде таблицы инциденций треугольников, а также допустимая погрешность аппроксимации.

Для удобства тестирования в качестве входных данных программы используются графические файлы в нескольких распространенных форматах. Изображение представляется как три прямоугольных сетки узлов одинакового размера, значение аппроксимируемой функции на каждой сетке задается уровнем яркости соответствующей компоненты (красный, синий, зеленый) цвета в отвечающем ему пикселе исходного файла. На полученных таким образом сетках строятся триангуляции, изоморфные описанной выше, затем каждая сетка укрупняется независимо. Для каждой компоненты строится своя собственная аппроксимация (в данном случае аппроксимация Куранта).

Функцией оценки погрешности получаемой аппроксимации взят косинус угла между нормалью к плоскости, задаваемой курантовской аппроксимацией в укрупненном треугольнике, и нормалью к плоскости сетки.

Выходные данные программы – таблица инциденций укрупненных триангуляций и построенные на них курантовские аппроксимации. В программе также реализован вывод созданного на приближенных значениях изображения в формат png.

Следует отметить, что в общем случае алгоритм не зависит от индексации узлов сетки и координат узлов. Таким образом, допускается работа с непрямоугольными и криволинейными правильными триангуляциями.

7. Обсуждение результатов. В таблице представлены результаты работы алгоритма для 3 и 10 последовательных укрупнений. В ней даны следующие колонки:

1) $size$ – размер исходного изображения (в пикселах);

2) V_0 – количество вершин, используемых в исходной триангуляции;

3) V_1 – количество вершин, применяемых в укрупненной триангуляции (в процентах от исходного количества вершин);

4) T_0 – количество треугольников исходной триангуляции;

5) T_1 – количество треугольников укрупненной триангуляции (в процентах от исходного количества треугольников).

Результаты работы алгоритма для различного количества укрупнений

$size$	V_0	$V_1, \%$	T_0	$T_1, \%$
3 укрупнения				
8×8	64	78	98	71
16×16	256	69	450	65
32×32	1024	44	1922	40
64×64	4096	27	7938	25
128×128	16 384	20	32 258	19
256×256	65 536	16	130 050	15
512×512	262 144	14	522 242	14
10 укрупнений				
8×8	64	78	98	71
16×16	256	69	450	65
32×32	1024	44	1922	40
64×64	4096	26	7938	24
128×128	16 384	15	32 258	14
256×256	65 536	7	130 050	6
512×512	262 144	6	522 242	6

На представленных результатах видно, что алгоритм позволяет выбросить из обработки значительную (более 90% на модельных примерах) часть узлов сетки с сохранением требуемой точности построенной на укрупненной сетке аппроксимации.

8. Заключение. В статье рассмотрен алгоритм, позволяющий проводить локальное укрупнение правильной триангуляции на двумерной плоскости с сохранением правильности, а также приведены некоторые результаты работы программы, реализующей этот алгоритм. Вместе с тем необходимо дополнительно исследовать эффективность работы алгоритма на различных наборах данных с использованием разных функций оценки получаемой погрешности.

Литература

1. *Демьянович Ю. К., Романовский Л. М.* Локальное укрупнение триангуляции и двумерные сплайн-вэйвлеты // Труды Всерос. науч. конференции по проблемам информатики. СПб.: ВВМ, 2012. С. 177–182.

2. *Демьянович Ю. К., Зимин А. В.* Аппроксимации курантова типа и их вэйвлетные разложения // Проблемы математического анализа. СПб.: Изд-во С-Петерб. ун-та, 2008. С. 3–22.

3. *Малла С.* Вейвлеты в обработке сигналов / пер. с англ. Изд. 2-е. М.: Мир, 2005. 671 с. (*Malla S.* Wavelets in signal processing.)

References

1. Demianovich U. K., Romanovskii L. M. *Lokalnoe ykrypnenie triangulacii i dvumernue splain-veivlety* (Local integration and two-dimensional triangulation spline wavelets). *Trydu vserossiyskoi naychnoi conferencii po problemam informatici.* St. Petersburg: VVM, 2012, pp. 177–182.

2. Demianovich U. K., Zimin A. V. *Approksimacii kyrantova tipa i ih veivletnye razlojenia* (Kurantova type approximations and wavelet decomposition). *Problemy matematicheskogo analiza.* St. Petersburg: Izd-vo S-Peterb. un-ta, 2008, pp. 3–22.

3. Malla S. *Veivlety v obrabotke signalov* (Wavelets in signal processing). Per. s angl. Izd. 2-e. Moscow: Mir, 2005, 617 p.

Статья рекомендована к печати проф. Л. А. Петросяном.

Статья поступила в редакцию 3 апреля 2013 г.