

**В.И. Солодка***Одесская национальная академия связи им. А.С. Попова, г. Одесса.***СРАВНИТЕЛЬНЫЙ АНАЛИЗ СЕТОЧНЫХ МЕТОДОВ С ПОМОЩЬЮ ЗАДАЧ ТРИАНГУЛЯЦИИ**

*В статье рассматриваются различные алгоритмы визуализации трехмерных объектов на эквидистантной сетке. Проводится исследование и описание методов визуализации, показываются преимущества и недостатки существующих методов.*

**Ключевые слова:** задача триангуляции, сеточные методы, трехмерные объекты.

**В.І. Солодка****ПОРІВНЯЛЬНИЙ АНАЛІЗ СІТКОВИХ МЕТОДІВ ЗА ДОПОМОГОЮ ЗАДАЧ ТРИАНГУЛЯЦІЇ**

*У статті розглядаються різні алгоритми візуалізації тривимірних об'єктів на еквідистантній сітці. Проводиться дослідження і опис методів візуалізації, показуються переваги і недоліки існуючих методів.*

**Ключові слова:** завдання тріангуляції, сіткові методи, тривимірні об'єкти.

**V. Solodka****COMPARATIVE ANALYSIS OF NET METHODS BY TASKS OF TRIANGULATION**

*The In the article the different algorithms of visualization of three-dimensional objects are examined on an ekvidistantnoy net. Research and description of methods of visualization is conducted, advantages and lacks of existent methods are shown.*

**Keywords:** task of triangulation, net methods, three-dimensional objects.

Данная статья посвящена сравнительному анализу алгоритмов визуализирующих заданную поверхность с помощью аппроксимации её треугольниками. Это так называемая задача триангуляции. Проблема визуализации поверхности, заданной различными способами возникает во многих областях математики, физики, медицины, телевидении.

Под задачей триангуляции понимается визуализация поверхности заданной с помощью функции от трех аргументов и фиксированного значения этой функции - уровня.

$$\{(x, y, z) / f(x, y, z) = c\},$$

где  $f(x, y, z)$  - это заданная функция, а  $c$  - заданный уровень.

Множество точек, удовлетворяющее этой формуле, и есть искомая поверхность. Однако удобнее восстанавливать не саму поверхность, а поверхность аппроксимирующую искомую с помощью треугольников. Такой способ визуализации называется триангуляцией.

При решении задачи визуализации большинство задач функция задается таблично

на сетке с эквидистантным шагом или имеет явное отображение, описываемое заданной формулой.

Но могут возникать задачи, в которых нет явно заданного отображения, или таблица значений задана на сетке с неэквидистантным шагом. Такие задачи могут возникать во многих приложениях, например, в задаче реконструкции трехмерной структуры с помощью множества контуров- «срезов» (в медицинских исследованиях). В таких задачах предлагается использовать следующий алгоритм действий [3]: поверхность  $S$ , заданная выборкой  $X$ , аппроксимируется касательными плоскостями, проходящими через каждую точку выборки  $X$ . Затем искомая функция, задающая поверхность, считается следующим образом: для каждой точки  $P$  пространства  $R$  функция в этой точке равна расстоянию до ближайшей касательной плоскости, взятому со знаком «+», если точка находится внутри объема, ограниченного построенными плоскостями, или со знаком «-», если точка находится вне этого объема. Затем проводится триангуляция поверхности, заданной с помощью полученной функции.

Рассмотрим методы решения задачи триангуляции. Ячеечные методы. В методах такого типа происходит разбиение области триангуляции на ячейки – параллелепипеды [1] или треугольные пирамиды. Далее производится триангуляция поверхности в каждой ячейке отдельно. Причем каждая ячейка триангулируется одним из заданных ранее способов, т.е. значения координат для треугольников просто «подставляются» из заранее заданной таблицы.

Для применения методов этого типа необходимо задать допустимую ошибку аппроксимации, на основе которой выбрать размер ячейки - треугольной пирамиды. После этого с помощью уже известных таблиц триангуляции получить искомое множество треугольников. При этом процедура триангуляции каждой ячейки сводится к анализу значений функции в вершинах этой ячейки – другими словами, определяется, какие вершины лежат «внутри» поверхности, а какие – «снаружи». На основе этого можно сделать вывод о достаточности определения функции только в вершинах ячеек.

Наиболее известные ячейчные алгоритмы: метод Канейро [2], метод «МТ6», метод Скалы [4], метод «Марширующих кубов» [1].

Метод предиктора-корректора (predictor-corrector). Методы из этого класса основаны на добавлении к уже имеющемуся множеству точек триангуляции ещё одной, лежащей на касательной плоскости к заданной функции (это положение предиктора (predictor) – предсказанное) и затем передвижению её до визуализируемой поверхности (это положение корректора (corrector) – скорректированное).

При использовании методов из этого класса, необходимо знать значение функции во всех точках пространства и найти хотя бы одну точку, принадлежащую искомой поверхности. Метод заключается в «наращивании» треугольников – на каждой итерации метода к уже существующему множеству треугольников добавляется еще один, построенный на ребре крайнего треугольника и предсказанной (а затем скорректированной по кривизне поверхности) точки на поверхности.

Мозаичные методы (pre-tessellation methods & particle-based methods). Суть таких методов заключается в разбиении искомой поверхности на части дальнейшей их триангуляции. Разбиение на части в pre-tessellation методах подразумевает разбиение поверхности на «примитивные» поверхности – фрагменты сфер и плоскостей. Разбиение на части в методах из плеяды particle-based – менее «интеллектуально» - ищутся только фрагменты плоскостей. При этом возникает проблема «соединения» уже

«протриангулированных» частей. Чаще всего этот процесс сводится к подбору локальных по Делоне треугольников, соединяющих части искомой поверхности.

Определение: треугольник локален по Делоне, если его самая маленькая сфера ограничения не содержит никакой другой точки триангуляции, которая имеет ту же самую поверхностную ориентацию.

Перечислим достоинства и недостатки методов триангуляции.

Основа методов первого типа – независимая триангуляция каждой ячейки с помощью таблиц триангуляции – является одновременно их сильной и слабой стороной. Высокая скорость работы этих методов делает их наиболее привлекательными по отношению к другим методам и дает возможность использовать их в интерактивных приложениях, но большим недостатком считается их относительная индифферентность к поведению функции вне выбранного множества точек. Другими словами это невозможность правильно визуализировать локальные искривления – масштаб треугольников всегда пропорционален размеру ячейки. Такие методы идеально подходят для визуализации трехмерных скалярных полей, заданных на регулярной сетке.

Методы второго и третьего типа применимы только при визуализации полей определенных в каждой точке той части пространства, которое нас интересует. Большим преимуществом таких методов можно считать их зависимость от локального искривления функции – в таких методах мелкие детали не «пропадут». Несмотря на сильную потерю в скорости по сравнению с методами первой группы и ограничения на дифференцируемость функции и связность поверхности, привлекают высоким «качеством» получаемой поверхности.

Заметное различие между ячейчными методами и методами второго и третьего типа заключается еще и в том, что методы первого типа часто довольно просты в реализации, и предоставляют возможность визуализации «нетривиально» заданных скалярных полей. Так, к примеру, создать регулярную сетку на основе нерегулярной значительно проще, нежели восстановить функцию в каждой точке пространства. Это же относится и к проблеме восстановления поверхности по «срезам» [5], возникающей в томографии.

Таким образом, учитывая постановку задачи, необходимо провести сравнительный анализ алгоритмов, относящихся к ячейчным методам по следующим критериям: скорость работы, ошибка аппроксимации, количество сгенерированных треугольников, качество генерируемых треугольников.

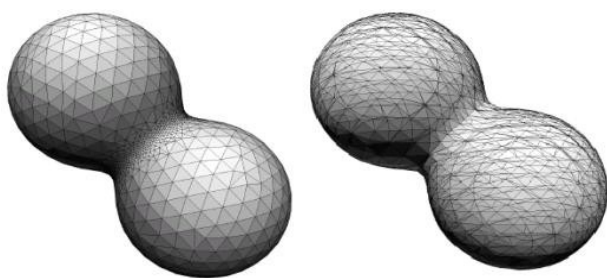


Рисунок 1 – Слева – триангуляція з урахуванням локальних искривлений; справа – без локальних искривлений. (Поверхність справа получена с помощью метода «Марширующих кубов»), слева – с помощью метода Стюарта (predictor-corrector).

Однако, рассматриваемые алгоритмы имеют одну и ту же основу. Поэтому скорость и ошибка аппроксимации различаются несущественно [4].

Таким образом, анализ алгоритмов достаточно провести по следующим критериям – количество треугольников, «качество» треугольников.

Алгоритм Скалы, относящийся к разряду ячеечных методов, был разработан, для визуализации трехмерных скалярных полей, заданных с помощью функции, определенной в каждой точке пространства. Однако, метод разбиения пространства на ячейки таков, что дает возможность использовать этот алгоритм для визуализации скалярных полей заданных на регулярной сетке.

Для разбиения пространства на ячейки метод Скалы использует узлы регулярной сетки, находящиеся в вершинах параллелепипеда, полученного тем же способом, что и в предыдущих рассмотренных методах, и дополнительную точку, находящуюся на пересечении диагоналей этого параллелепипеда. Значение функции в этой точке предлагается считать как линейную интерполяцию значений функции в вершинах параллелепипеда. Для каждого параллелепипеда, полученного из узлов регулярной сетки, строится ячейка способом. При таком разбиении для каждой ячейки используются «срединные» точки «соседних» параллелепипедов.

Алгоритм «МТ6» был предложен Гуезеком [3] как альтернатива алгоритму Канейро. Основное отличие этих двух методов в том, что для алгоритма «МТ6» отпадает необходимость смены шаблонов с прямого на зеркальный и обратно. Это достигается путем симметричного разбиения параллелепипеда, показанного на рис. 2.

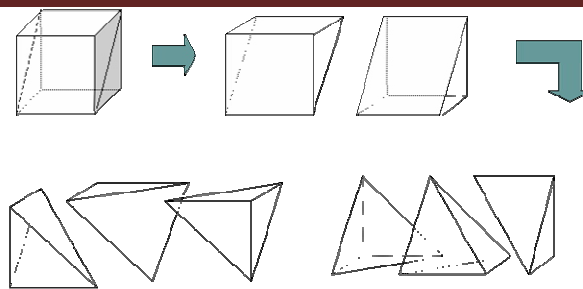


Рисунок 2 – Симметричное разбиение параллелепипеда

Алгоритм «Марширующие кубы» Алгоритм «Марширующие кубы», предложенный Лоренсеном [1], можно разбить на два этапа:

1. Разбиение области  $G$  пространства  $R^3$  на конечное множество ячеек, поиск ячеек пересекаемых искомой поверхностью.

2. Аппроксимация поверхности в найденных ячейках.

Алгоритм Канейро [2], основанный на разбиении пространства на треугольные пирамиды, как и алгоритм «Марширующие кубы», состоит из двух этапов:

1. Разбиение пространства на конечное множество ячеек, поиск ячеек пересекаемых искомой поверхностью.

2. Аппроксимация поверхности в найденных ячейках.

Для сравнения методов были использованы следующие критерии:

- Количество генерируемых треугольников
- «Качество» генерируемых треугольников

Сравнение производилось следующим образом: на ввод алгоритмам подавались тестовые поверхности, заданные на регулярной сетке  $30 \times 30 \times 30$ . Для каждой поверхности просчитывалось количество треугольников, вектор качества треугольников и средняя мера правильности треугольника.

Вектор качества треугольников определяется следующим образом:

$$BKT = \{a_1, a_2, \dots, a_{10}\}, a_i \in [0, 100], \sum_i a_i = 100.$$

При этом,  $i$ -ый элемент вектор качества треугольников определяется как процентное содержание треугольников с мерой качества  $V$ , во множестве сгенерированных треугольников, где  $V$  удовлетворяет следующему неравенству:

$$\frac{i-1}{10} \leq V < \frac{i}{10}, i \neq 10,$$

$$V \geq 0,9, i = 10.$$

Средняя мера правильности треугольников подсчитывалась как среднее арифметической мер правильности всех треугольников.

При анализе полученных результатов можно сделать следующие выводы:

Алгоритм Скалы генерирует неоправданно большое количество треугольников (к примеру, при визуализации поверхности тора этот алгоритм произвел в 5 раз больше треугольников, чем алгоритм «Марширующие кубы»).

Алгоритм МТ6 показал на всех тестовых поверхностях результаты хуже, чем алгоритм Канейро (для каждой тестовой поверхности количество генерируемых треугольников алгоритмом МТ6 было выше, чем количество генерируемых треугольников алгоритмом Канейро, а средняя мера качества ниже).

Алгоритм «Марширующие кубы» генерирует значительно меньшее количество треугольников, чем другие алгоритмы.

Таким образом, можно сказать, что алгоритм «Марширующие кубы» имеет лучшие показатели качества по выбранным критериям. Однако, Дёрст (Durst) в своей работе показал топологическую неточность построенных поверхностей с помощью алгоритма «Марширующие кубы».

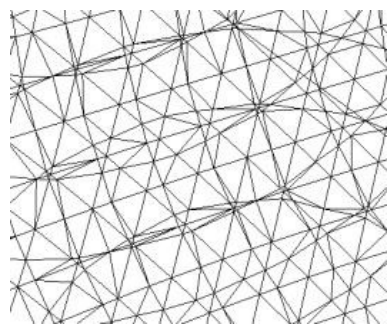
Решения этой проблемы, основаны на изменении шаблонов триангуляции и попытке

«предугадать» (отсюда пошло название – asymptotic decider) как на самом деле себя ведет поверхность в ячейке. Однако такой подход в несколько раз увеличивает количество треугольников, генерируемых алгоритмом «Марширующие кубы». К тому же, такой алгоритм уже нельзя считать интерактивным [6].

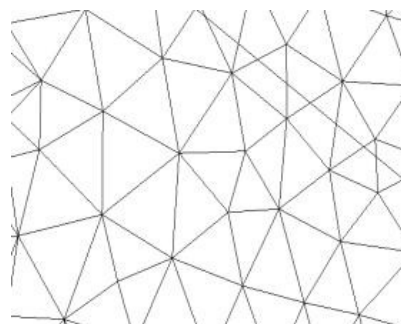
Несмотря на существование топологической неточности в генерируемой поверхности, алгоритм «Марширующие кубы» широко используется на практике, т.к. вероятность проявления ошибок такого рода достаточно мала. К примеру, при визуализации тестовых поверхностей топологическая неточность ни разу не проявилась.

Таким образом, при решении прикладных задач на интерактивную визуализацию трехмерных скалярных полей заданных на эквидистантной сетке, лучше всего использовать алгоритм Канейро, в том случае, если важна топологическая точность получаемой поверхности. В противном случае, целесообразнее использовать алгоритм «Марширующие кубы».

Однако показатели качества алгоритма «Марширующие кубы» далеки от тех результатов, которые дают не интерактивные алгоритмы.



а



б.

Рисунок 3 – Фрагмент сферы: а – полученный в результате работы алгоритма «Марширующие кубы», б – после не интерактивного преобразования того же множества треугольников

#### Список использованных источников:

1. William E. Lorensen, Harvey E. Cline, CG vol.21, no.4, – July 1987
2. Bernardo P. Carneiro, Arie E. Kaufman, SIGGRAPH'96, pp. 205 – 210.
3. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. SIGGRAPH'92 proceedings, 26(2), pp. 71 – 78.
4. Vaclav Skala, Conference on Scientific Computing – 2000. pp. 368 – 378. [http://www.emis.de/journals/AMUC/\\_contributed/algo2000/skala.pdf](http://www.emis.de/journals/AMUC/_contributed/algo2000/skala.pdf)

5. Paolo Cignoni, IEEE Transaction on visualization and CG, vol.3, no.2 April-June 1997

6. Andre Guezic, IEEE Transactions on Visualization and Computer Graphics, Vol. 1, Issue 4, pp. 328 – 342, – December 1995

**Рецензент:** д.т.н., профессор Гофайзен О.В., Одесская национальная академия святы им. А.С. Попова, Одесса