Гибридный рекурсивно - инкрементный алгоритм построения триангуляции Делоне

Синицын С.И.
Нижний Новгород

Аннотация

Изложен алгоритм построения триангуляции Делоне. Его особенность заключается в специально подобранной схеме инкрементного включения новых треугольников. Использование подобной схемы позволяет при каждом проходе алгоритма разбивать остающееся количество точек на два приблизительно одинаковых независимых множества, что обеспечивает рекурсивность алгоритма.

Ключевые слова

Триангуляция Делоне, диаграмма Вороного, рекурсивно - инкрементный алгоритм, вычислительная геометрия, компьютерная графика, визуализация научных результатов.

1. ВВЕДЕНИЕ

Триангуляция Делоне (ТД) и двойственная ей диаграмма Вороного (ДВ)– это ключевые объекты вычислительной геометрии [1]. С ее помощью решаются разнообразные задачи на близость (выделим построение минимального остовного дерева и евклидова дерева Штейнера). ТД используется при визуализации топографической местности в геоинформационных системах. Она может использоваться при построении сетки для численного анализа и визуализации результатов математического и натурного моделирования и т.д.

Напомним известные сведения о триангуляции Делоне.

Триангуляцией набора S, состоящего из N точек, называется сетка примыкающих друг к другу треугольников, таких, что вершинами любого треугольника являются точки из набора S, и ребра треугольников не могут пересекаться в других точках, не принадлежащих S. Точки и ребра, принадлежащие выпуклой оболочке CH(S) называются граничными, остальные – внутренними. Обозначим общее количество внутренних точек через V. Любой набор точек, как правило, допускает несколько способов триангуляции. Но при этом существует замечательное свойство, которое заключается в том, что при любом способе триангуляции общее число треугольников остается постоянным и равным K= N+V-2.

Другое свойство заключается в том, что каждое из внутренних ребер принадлежит ровно 2 примыкающим друг к другу треугольникам. Ребро, принадлежащее выпуклой оболочке, входит в состав единственного треугольника.

Известны следующие схемы триангуляции: выпуклое обминание, «жадная», минимальная взвешенная, Делоне.

ТД к настоящему времени получила наибольшее распространение в большинстве графических приложений. Для триангуляции Делоне существует несколько эквивалентных определений. Здесь приведем то, которое явным образом используется в описанном ниже алгоритме. Оно заключается в том, что если вокруг любого треугольника из набора K провести описанную окружность, то внутрь этой окружности не попадут другие точки набора S. Отсутствие попадания точки точно на любую описанную окружность приводит к единственному набору ТД.

2. ИЗВЕСТНЫЕ АЛГОРИТМЫ

Все известные алгоритмы для построения триангуляции Делоне можно разделить на два класса:

1. инкременционные. Они конструируют триангуляцию, начиная с любой внутренней точки или ребра границы, пошагово добавляя полученные новые треугольники по одному. Сложность их обычно оценивается величиной $O(N^2)$.
2. рекурсивные («разделяй и властвуй»). Эти алгоритмы рекурсивно разбивают заданное множество точек на поднаборы примерно одинакового размера с последующим слиянием полученных данных в необходимый результат. Эти алгоритмы могут достигать асимптотически оптимальную сложность, которая оценивается величиной O(N logN).

Рассмотрим более подробно инкрементный алгоритм, описанный в [2]. На каждой итерации алгоритм ищет новый треугольник, который подключается к текущей *границе.* Определим понятие *границы*, которое зависит от следующей схемы классификации ребер ТД в процессе работы алгоритма. Каждое ребро может быть *спящим*, *живым* или *мертвым*. Ребро считается *спящим*, если оно еще не обнаружено алгоритмом. Ребро *живое*, если оно ориентировано, обнаружено алгоритмом, но известен, только один примыкающий к нему слева треугольник. Ребро *мертвое*, если известны оба примыкающих к нему треугольника. *Замечание: ребрам выпуклой оболочки, первоначально ориентированным по часовой стрелке, может принадлежать только один треугольник. Однако, это не требует специальной граничной модификации алгоритма, т.к. при обработке их не находится точек, расположенных справа от живого граничного ребра для образования треугольника и он автоматически переводится в спящие. Ориентация такого ребра меняется на противоположную.*

Граница – это набор *живых* ребер. Вначале живым является единственное ребро, принадлежащее выпуклой оболочке, а все остальные ребра спящие. По мере работы алгоритма ребра из спящих становятся живыми, а затем мертвыми. На каждой итерации обрабатывается любое из ребер живой границы. Эта обработка заключается в том, что для всех точек, расположенных справа от живого ребра определяется такая, для которой при проведении описанной окружности по трем точкам (две принадлежат концам живого ребра, а третья - обрабатываемая) выполняется свойство ТД (внутри этой окружности нет других точек). Наиболее простой способ нахождения такой окружности связан с построением параметризованного срединного перпендикуляра к живому ребру и нахождения параметра точки пересечения его с любым другим срединным перпендикуляром рассматриваемого треугольника. Наименьшее значение получаемого параметра **t** (рис.1) при рассмотрении всех расположенных справа точек (и внутренних, и граничных) указывает на необходимую точку. Ей в данном случае является точка А. Этот алгоритм образно можно представить в виде надувания пузыря вправо от живого ребра, две точки которого касаются его концов.

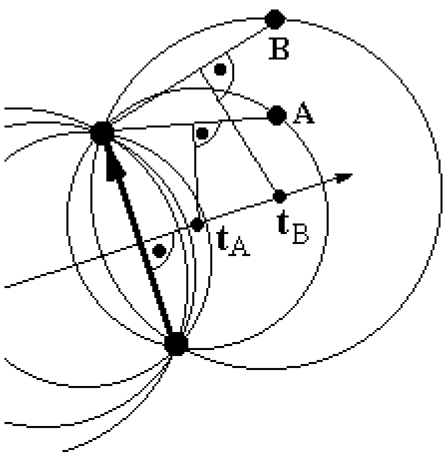

Рис.1. К определению лучшей точки для триангуляции Делоне ($t_A < t_B$).

Этот же метод мы будем использовать при определении нужной нам точки и в гибридном методе, описанном в п.3. Первая из встреченных точек и является искомой для перестроения *границы*. Такое перестроение заключается в том, что живое ребро становится мертвым, а к границе добавляются два новых ориентированных *живых* ребра. При этом ориентация вводится следующим образом:

- для ребра, соединяющего начало ставшего на данной итерации мертвым ребра и полученной точкой – ориентация от начала к точке;
- для ребра, соединяющего точку с концом ставшего на данной итерации мертвым ребра– ориентация от точки к концу;

Алгоритм заканчивает свою работу, когда исчерпывается список *живых* ребер *границы*. Т.к. при включении каждой новой точки приходится просматривать весь список оставшихся точек, то сложность данного алгоритма, оценивается $O(N^2)$.

Рассмотрим более подробно рекурсивный алгоритм построения диаграммы Вороного (ДВ), описанный в [1]. ДВ двойственна ТД в том смысле, что срединный перпендикуляр к любому ребру триангуляции Делоне является стороной области Вороного. Это означает, что один из построенных объектов (ТД или ДВ) однозначно определяет второй.

Общая схема рекурсивного алгоритма построения ДВ состоит в следующем.

Разделим множество S на 2 подмножества S_1 и S_2, использовав для этого медиану по оси абсцисс. Рекурсивно построим ДВ на S_1 и S_2. Построим ломаную σ, разделяющую S_1 и S_2. Удалим все ребра диаграммы для S_2, расположенные справа от σ и все ребра диаграммы для S_1, расположенные слева от σ. В результате получаем ДВ для множества S в целом. Наибольшая трудность при использовании данного алгоритма заключается в построении разделяющей ломаной σ. Авторы приводят достаточно сложный алгоритм такого построения и доказывают оптимальность его, указав на сложность O(NlogN).

Учитывая простоту понимания первого алгоритма и рекурсивное разделение второго предложим гибридный алгоритм.

Гибридный алгоритм

В качестве предобработки построим на множестве S выпуклую оболочку CH(S), каждое из ребер оболочки сделаем ориентированным с учетом обхода по часовой стрелке. Определим медианы по абсциссе X_M и ординате Y_M по формулам:

$$X_M = \left(\sum_{i=1}^{N} X_i\right) / N; \qquad Y_M = \left(\sum_{i=1}^{N} Y_i\right) / N$$

.

Проведем горизонтальную и вертикальную прямые X_M=const и Y_M=const. Найдем точки пересечения этих прямых с соответствующими ребрами выпуклой оболочки. Сравнив расстояния по вертикали и горизонтали и учитывая предположение о случайности набора точек, за *базовую* возьмем прямую, для которой расстояние между точками пересечения меньше. Это позволит быстрее разделить исходное множество на два подмножества. Дальнейшую схему покажем на примере горизонтальной *базовой* прямой (рис.2).

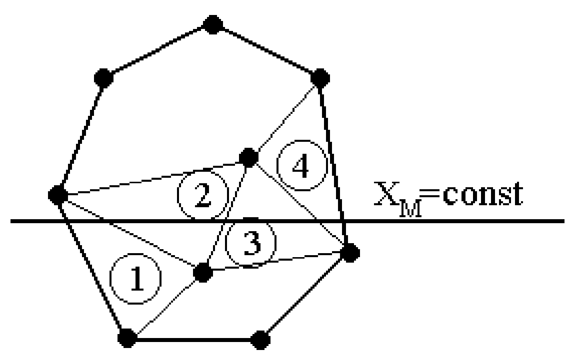

Рис. 2. Последовательность выделения разделяющей цепи выбранных треугольников.

Предварительно все непересекаемые ребра оболочки, расположенные выше X_M, заносим в *список ребер верхней оболочки*, а точки как внутренние, так и граничные, расположенные выше X_M в *верхний список точек*; непересекаемые и расположенные ниже X_M ребра оболочки заносим в *список ребер нижней оболочки*, а все точки расположенные ниже X_M помещаем в *нижний список точек*. В качестве первого исследуемого ребра используем левое ребро выпуклой оболочки такое, которое пересекает базовая прямая. По методу рис.1 находим подходящую точку, определив тем самым 1 треугольник. Если прямая X_M пересекает условную верхнюю сторону, как показано на рис. 2, то она используется для продолжения построения цепи разделяющих треугольников. Нижнее непересекаемое прямой X_M ребро добавляется в *список нижней оболочки* Совершенно аналогично, как это показано для треугольника 2 условно нижнее ребро используется для последующего определения треугольника 3, а верхнее непересекаемое прямой X_M ребро добавляется в с *список верхней оболочки.* Точки, участвующие в цепи разделяющих треугольников, исключаются из соответствующих *верхнего* и *нижнего списков точек*

Построение разделяющей цепи треугольников, что считается единым проходом алгоритма, заканчивается, когда второе ребро пересекаемого треугольника совпадает с правым ребром оболочки, пересекаемой прямой X_M. На рис.2 это случилось для треугольника 4. В результате первого прохода построена цепь разделяющих треугольников Делоне 1, 2, 3, 4 и подготовлены данные для следующих проходов алгоритма.

Рекурсивным окончанием разделения точек служит либо отсутствие оставшихся точек в условно верхнем или нижнем списке точек, либо наличие 3 оставшихся точек в таком списке, что образует последний треугольник. Ввиду очевидной простоты, можно предусмотреть модификацию окончания рекурсивного деления, если остаются 4 точки в списке.

ЗАКЛЮЧЕНИЕ

Гибридность предлагаемого алгоритма накладывает отпечаток и на оценку его временной сложности. По предварительным оценкам она должна находится где-то между O(NlogN) и $O(N^2)$. Это определяется тем, что в процессе построения разделяющей цепи треугольников мы фактически используем модификацию первого из рассмотренных нами известных методов. Но в результате такой модификации общее количество точек делится на независимые примерно равные множества, что позволяет говорить об асимптотически оптимальной сложности.

Естественно, что данный алгоритм целесообразно использовать на очень больших наборах заданных точек.

ЛИТЕРАТУРА

[1] Препарата Ф., Шеймос М. Вычислительная геометрия: Введение: Пер. с англ. – М.: Мир, 1989. – 478с.

[2] Ласло М. Вычислительная геометрия и компьютерная графика на C++: Пер. с англ. – М.: «Издательство БИНОМ», 1997. – 304с.

Сведения об авторе

СИНИЦЫН СЕРГЕЙ ИВАНОВИЧ – кандидат технических наук, доцент кафедры начертательной геометрии, машинной графики и САПР Нижегородского государственного архитектурно- строительного университета, тел. (8312)-30-54-00.
e-mail: graphics@cad.ngasu.sci-nnov.ru