

Метод слияния перекрывающихся триангуляций Делоне*

Л.М. Местецкий, М.Л. Готман

mestlm@mail.ru | gotman.maria@gmail.com

Россия, Московский государственный университет имени М.В.Ломоносова

Рассматривается задача слияния двух триангуляций Делоне, причем множества, на которых заданы исходные триангуляции, допускают полное перемешивание. Известные методы имеют сложность построения триангуляции $O(n \log n)$ или предполагают пространственное разделение исходных триангуляций, тогда слияние производится за $O(n)$. Практическое применение данного алгоритма состоит в сравнении 3D поверхностей, заданных на нерегулярных сетках, при котором осуществляется подгонка пары поверхностей друг к другу, на каждой итерации подгонки выполняется слияние триангуляций. Сложность слияния сильно влияет на эффективность решения задачи. Предложенный алгоритм имеет сложность $O(n)$ по числу точек в худшем случае для слияния перекрывающихся триангуляций Делоне.

Ключевые слова: триангуляция Делоне, слияние триангуляций, неразделенные триангуляции.

1. Введение

Задача построения триангуляции Делоне (ТД) n сайтов-точек имеет нижнюю оценку вычислительной сложности $O(n \log n)$. В случаях, когда имеется некоторая дополнительная информация, эта оценка может быть улучшена. В частности, представляет интерес задача слияния двух триангуляций, когда исходное множество сайтов состоит из двух непересекающихся подмножеств S_1 и S_2 , и триангуляции Делоне $Del(S_1)$ и $Del(S_2)$ исходных множеств уже построены.

В такой постановке задача слияния триангуляций Делоне рассматривалась на конференции Графикон в 2004 году [1]. Недостатком той работы является сложный этап поиска стартеров. В настоящей работе этот шаг значительно упрощен, были доработаны структуры данных, появилась более четкая математическая терминология.

Наш интерес к этой задаче связан с построением алгоритма для вычисления расстояния в метрическом пространстве функций двух переменных, заданных на конечных нерегулярных множествах точек. Эта задача возникает, в частности, при анализе 3D поверхностей человеческих лиц, полученных в результате пространственного сканирования [2]. Предложенная модель определяет расстояние между такими функциями на основе построения общей ТД. При этом расстояние между моделями лиц определяется как минимальное по всем движениям множеств S_1 и S_2 на плоскости. Таким образом, осуществляется подгонка пары поверхностей друг к другу. На каждой итерации подгонки выполняется слияние триангуляций.

2. Терминология и постановка задачи

Пусть на евклидовой плоскости задано S — множество из $n \geq 3$ точек-сайтов, не все из которых лежат на одной прямой.

Триангуляцией конечного множества точек S называется планарный граф с вершинами из S , все внутренние области которого являются треугольниками. Триангуляция называется выпуклой, если минимальный многоугольник, охватывающий все ее треугольники, будет выпуклым.

Окружность называется пустой, если она не содержит внутри себя сайтов. Окружность, проходящая через сайт, называется инцидентной этому сайту. Ребром Делоне называется ребро, вершины которого имеют общую пустую инцидентную окружность. Гранью Делоне называют грань, вершины которой имеют общую пустую инцидентную окружность.

Триангуляцией Делоне (ТД) $Del(S)$ множества точек S называется выпуклая триангуляция, все грани которой являются гранями Делоне. Эквивалентное определение — выпуклая триангуляция, все ребра которой являются ребрами Делоне.

Пучком сайта $p \in S$ называют множество ребер триангуляции, инцидентных сайту p .

Задача слияния двух перекрывающихся триангуляций Делоне ставится следующим образом. Даны два конечных линейно неразделимых множества сайтов B и W (для удобства изложения будем считать, что сайты раскрашены в два цвета: множество n_1 черных сайтов B и множество n_2 белых сайтов W , $n = n_1 + n_2$), а также их триангуляции Делоне $Del(B)$ и $Del(W)$. Нужно построить триангуляцию Делоне на объединенном множестве сайтов $Del(B \cup W)$ (рис. 1).

3. Описание метода

В процессе построения результирующей триангуляции Делоне $Del(B \cup W)$ часть ребер переходит из исходных ТД без изменений, при этом оставшаяся часть ребер исходных триангуляций перестает удовлетворять условию Делоне на объединенном множестве точек, и должна быть разрушена. Также происходит построение новых разноцветных ребер, то есть таких, что одна вершина ребра принадлежит белой триангуляции, вторая — черной.

Работа опубликована по гранту РФФИ №16-07-20482.

Латентным ребром называется разноцветная пара сайтов, для которой выполнено условие Делоне по всему множеству сайтов $B \cup W$. Дефектным ребром называется одноцветная пара сайтов, образующая ребро Делоне в

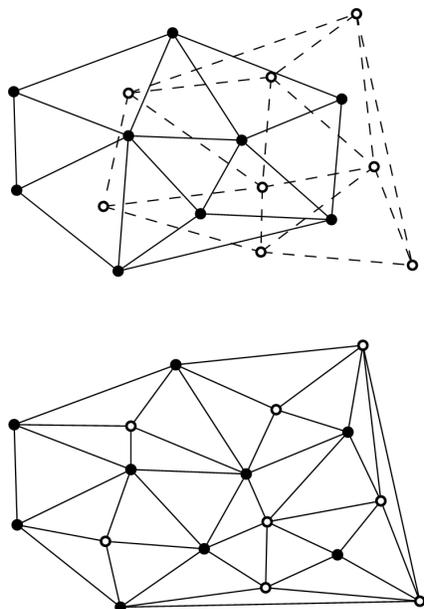


Рис. 1: Исходные триангуляции и объединенная триангуляция.

$Del(B)$ или $Del(W)$, для которой не выполнено условие Делоне на множестве $B \cup W$. Открытым ребром называется латентное ребро, имеющее не более одной инцидентной грани, не лежащее в выпуклой оболочке ТД, либо не имеющее инцидентных граней и лежащее в выпуклой оболочке ТД. Швом называется максимальная связная последовательность латентных ребер. Стартером называется открытое ребро, от которого начинается построение шва.

3.1 Структура данных

Существуют различные структуры данных для хранения триангуляции Делоне. Переход от одной структуры данных к другой осуществляется за $O(n)$. В нашей задаче будут часто производиться коррекции пучков сайтов, поэтому операция добавления и удаления ребер из пучка должна производиться очень эффективно. Нам удобно использовать структуру данных «Узлы с соседями» (рис. 2). Такая структура для каждого сайта хранит его координаты на плоскости и список номеров смежных сайтов в обходе против часовой стрелки.

3.2 Условие существования латентного ребра

Лемма 1. Сайт имеет инцидентное латентное ребро \Leftrightarrow существует инцидентная ему окружность, внутри которой нет сайтов одного с ним

цвета, но есть сайты противоположного цвета на ней или внутри нее.

Согласно лемме 1, если существует сайт A и инцидентная ему окружность O (рис. 3), удовлетворяющая условию леммы, то поиск стартера сводится к перебору сайтов $D_1, D_2, \dots, D_k, k \geq 1$, отличного от A цвета, внутри этой окружности и выбору сайта D_{min} , такого что инцидентная A и D_{min} окружность имеет с окружностью O общую касательную в точке A и минимальный радиус из всех таких окружностей, инцидентных A и D_i . Тогда AD_{min} является новым латентным ребром.

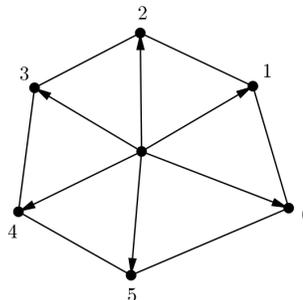


Рис. 2: Структура данных «Узлы с соседями».

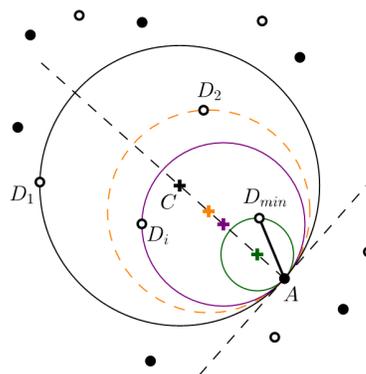


Рис. 3: Условие существования латентного ребра.

3.3 Поиск первого стартера

Будем говорить, что сайт A лежит левее сайта B , если A предшествует B при лексикографическом упорядочивании сайтов по возрастанию координат (x, y) .

Пусть B и W множества сайтов исходных триангуляций. Найдем сайты B_{min} и W_{min} , являющиеся самыми левыми в соответствующем множестве точек B и W . Не ограничивая общности, будем считать, что B_{min} лежит левее, чем W_{min} (рис. 4). Рассмотрим окружность O , инцидентную сайтам B_{min} и W_{min} с центром на луче, параллельном оси Ox , выходящим из точки W_{min} влево. Окружность O не содержит ни одного сайта из W , потому что лежит левее самого левого сайта этого множества W_{min} , и содержит сайт B_{min} на границе окружности. Если

окружность не пустая, то часть сайтов из B попадет внутрь этой окружности. Тогда точка W_{min} и окружность O удовлетворяют условиям леммы 1. Это дает возможность найти первый стартер (на рисунке изображен красным цветом).

3.4 Выявление дефектных ребер

Пусть найдена пара сайтов A и B , образующая латентное ребро, ещё не включенное в объединенную триангуляцию.

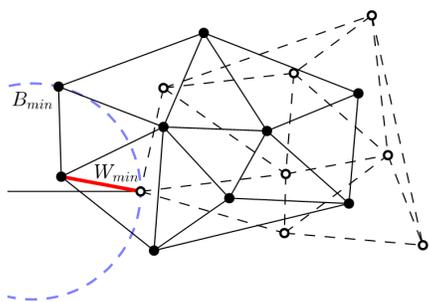


Рис. 4: Поиск первого стартера.

Присоединение ребра AB к пучкам сайтов-вершин A и B может привести к нарушению условия Делоне для некоторых ребер из этих пучков, то есть часть ребер окажутся дефектными. Выявление дефектных ребер осуществляется сразу после присоединения нового латентного ребра на основе углового критерия:

Лемма 2. (Угловой критерий) Пусть S — множество сайтов, для пары $A, B \in S$ выполнено условие Делоне, тогда для любых других двух сайтов $C, D \in S$, лежащих по разные стороны от AB , справедливо: $\angle ACB + \angle ADB \leq 180^\circ$; $\angle ABD \geq \angle ACD$.

Пусть AC_1 и AC_2 одноцветные ребра пучка A , что сайты B, C_1 и C_2 идут в порядке обхода против часовой стрелки в пучке сайта A (рис. 5). Для ребра AC_1 нарушено условие Делоне, если $\angle AC_2C_1 + \angle ABC_1 > 180^\circ$ (лемма 2). В этом случае ребро AC_1 становится дефектным, и удаляется из объединенной триангуляции, а ребро AC_2 подвергается аналогичной проверке. Если же для ребра AC_1 условие Делоне выполнено, то ребро AC_1 сохраняется в объединенной триангуляции, проверка условия Делоне для остальных ребер не производится.

Аналогичным образом выполняется коррекция пучка сайта B в направлении обхода по часовой стрелке. После этого производим коррекцию ребра AB в противоположном направлении, то есть точки B, C_1 и C_2 идут в порядке обхода по часовой стрелке в пучке сайта A , пучок сайта B корректируем против часовой стрелки. Таким образом, пучки сайтов A и B будут скорректированы как по часовой стрелке, так и против нее. На этом присоеди-

единение нового латентного ребра AB заканчивается.

Все дефектные ребра сохраняются в отдельный список *dumping* вместе с точкой, которая нарушает условие леммы 2, то есть при удалении ребра AC_1 сохраним точку B , если $\angle ABC_1 > \angle AC_2C_1$, и точку C_2 в противном случае. Эта информация понадобится нам при дальнейшем поиске стартеров.

3.5 Поиск латентных ребер

Пучки, для которых выполнена описанная коррекция, будем называть правильными. Рассмотрим алгоритм поиска новых латентных ребер объединенной ТД:

Лемма 3. Пусть AB открытое ребро, пучки сайтов A и B являются правильными. Пусть C следует за точкой B в пучке сайта A , точка D предшествует точке

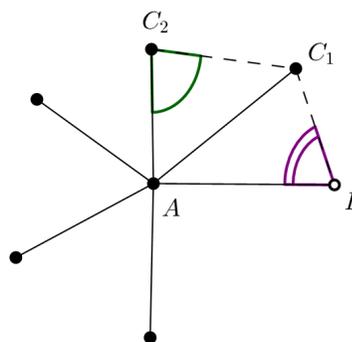


Рис. 5: Удаление дефектных ребер.

A в пучке сайта B , тогда если $\angle ACB > \angle ADB$, то BC является новым латентным ребром, иначе AD является новым латентным ребром Делоне (рис. 6).

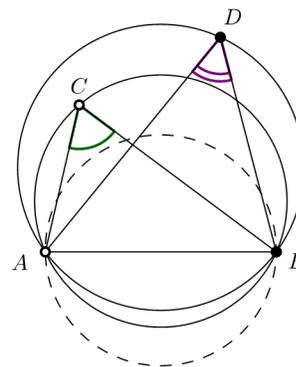


Рис. 6: Построение латентных ребер.

В процессе построения латентных ребер в любой момент времени имеется не более двух открытых ребер (стартер и новое латентное ребро). Пока очередное латентное ребро является открытым, продолжаем построение последующих латентных ребер

бер, и выявление дефектных ребер, соответствующих новому латентному ребру. Если очередное построенное ребро перестало являться открытым, то оно могло совпасть со вторым открытым ребром, в этом случае процесс построения шва считаем законченным. Либо оказалось ребром выпуклой оболочки, тогда нужно продолжать построение шва в противоположную сторону от стартера (оставшегося открытого ребра), пока очередное ребро не будет являться ребром выпуклой оболочки. На этом процесс построения шва считаем законченным, для построения последующих швов требуется найти следующий стартер.

3.6 Поиск стартеров на основе минимальных остовных деревьев

Минимальным остовным деревом (МОД) триангуляции Делоне называется ее связный подграф, имеющий наименьшую суммарную длину ребер. Пусть на плоскости задано N точек. Евклидовым МОД (ЕМОД) называется связный подграф, вершинами которого являются все N точек, суммарная длина всех ребер которого минимальна (рис. 7). Известно, что МОД ТД является ЕМОД для множества сайтов ТД [3, стр. 227, 277]. Задача поиска последующих стартеров основана на использовании минимальных остовных деревьев исходных триангуляций Делоне. МОД исходных триангуляций вычисляются один раз, поэтому это действие можно отнести к этапу предобработки, предшествующему непосредственному слиянию исходных триангуляций. Теоретическая оценка трудоемкости задачи построения минимального остовного дерева составляет $O(n \log n)$. Известно, что МОД является подграфом триангуляции Делоне, и на ее основе МОД может быть построен за линейное время. Такой вычислительной сложностью обладает алгоритм Черитона-Тарьяна [3, стр. 226-230].

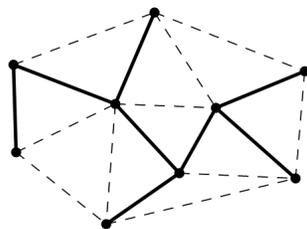


Рис. 7: ЕМОД триангуляции Делоне.

Поиск последующих стартеров основан на том, что если при удалении дефектных ребер нарушается связность исходных триангуляций, то среди дефектных ребер обязательно окажется ребро МОД этой ТД. *Мостами* будем называть дефектные ребра исходных триангуляций, входящие в МОД. Будем называть окружность, диаметром которой яв-

ляется ребро МОД, *окружностью влияния ребра МОД*.

Лемма 4. *Окружность влияния ребра МОД является пустой окружностью ТД.*

В *dumping* хранятся все дефектные ребра, и для каждого ребра хранится точка, нарушающая критерий леммы 2 для этого ребра. Пусть AB — мост (не ограничивая общности будем считать его белым). По лемме 4 в окружности влияния нет вершин белого цвета, зато есть вершины черного цвета. По лемме 1 среди всех черных вершин, попавших в окружность влияния AB , нужно найти ту, которая образует пустую окружность с сайтом B . При разрушении AB мы запомнили черную вершину, C , которая нарушала условие Делоне для моста AB , то есть вершина C попадает в окружность влияния AB . Поиском в ширину, начиная с точки C , среди точек $D_1, D_2, \dots, D_k, k \geq 1$, попавших в окружность влияния AB , ищем искомую точку D_{\min} .

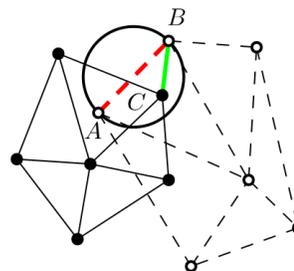


Рис. 8: Поиск последующего стартера.

На рисунке 8 AB — разрушенный мост, точка C нарушает условие Делоне для AB . В окружность влияния AB попала только одна точка C , поэтому ребро BC становится новым стартером.

3.7 Общая структура алгоритма

Таким образом, предложенный метод можно представить в виде последовательности шагов:

1. Поиск первого стартера.
2. Построение латентных ребер, принадлежащих одному шву, и удаление дефектных ребер.
3. Поиск последующего стартера.
4. Если стартер найден, перейти к шагу 2, иначе закончить построение объединенной триангуляции.

4. Заключение

В данной работе был предложен вычислительно эффективный алгоритм объединения двух перекрывающихся триангуляций Делоне. Предложенный метод позволяет эффективно производить сравнение 3D поверхностей, при этом построение двух исходных ТД и МОД производится один раз,

и относится к этапу предобработки, объединение триангуляций на каждом шаге подгонки поверхностей будет производиться за $O(n)$.

Алгоритм был реализован на языке Python. Вычислительный эксперимент подтверждает теоретическую оценку сложности алгоритма $O(n)$.

Литература

- [1] Л. М. Местецкий and Е. В. Царик, “Триангуляция Делоне: рекурсия без пространственного разделения точек,” in Труды 14-й Международной конференции по компьютерной графике и зрению ГрафиКон’2004. 2004, GraphiCon, pp. 267–270, МГУ Москва.
- [2] N. Dyshkant, “Comparison of point clouds acquired by 3d scanner,” Lecture Notes in Computer Science, vol. 7749, pp. 47–58, 2013.
- [3] Franco P. Preparata and Michael I. Shamos, Computational Geometry: An Introduction, SpringerVerlag New York, Inc., New York, NY, USA, 1985.