Министерство образования Российской Федерации

Удмуртский государственный университет

Математический факультет

Кафедра алгебры и топологии

Курсовая работа на тему

"Визуализация алгоритма Грэхема построения выпуклой оболочки с помощью JavaScript"

Выполнил:
студент гр. 010200-33
Пучков И. А.

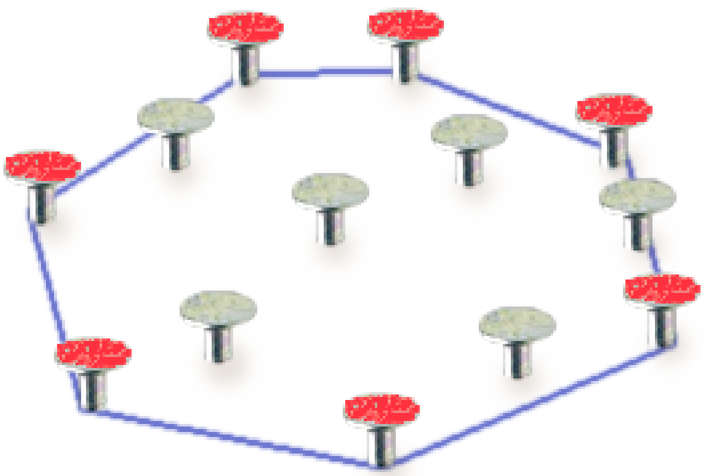
Научный руководитель:
Ст. преподаватель
Бастрыков Е. С.

Ижевск 2014

Содержание

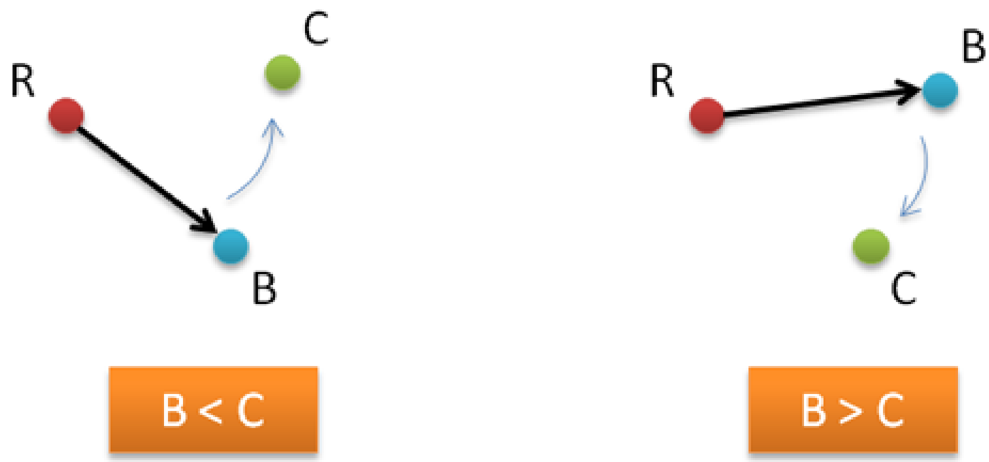
1 Определение выпуклой оболочки.

Выпуклой оболочкой множества X называется минимальное выпуклое множество, содержащее X. Множество X в векторном пространстве называется выпуклым, если оно содержит вместе с двумя любыми точками соединяющий их отрезок. Минимальной выпуклой оболочкой (далее МВО) называется выпуклая оболочка минимальной длины. Наиболее частым примером выпуклой оболочки приводится деревянная доска, в которую вбито (не до конца, естественно) много гвоздей. Если взять резинку (или веревку) и натянуть её вокруг гвоздей, то получится выпуклая оболочка, которая окружает все вбитые гвозди, но касается только некоторых. Те гвозди, которых она касается, составляют выпуклую оболочку для всех гвоздей.

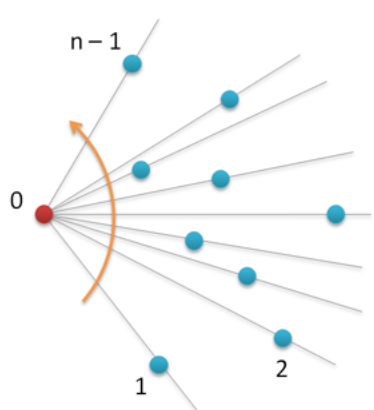


2 Описание алгоритма Грэхема.

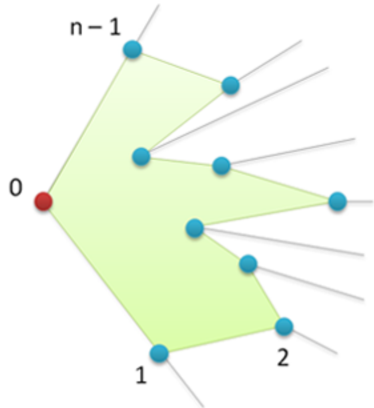
Существует множество методов построения выпуклой оболочки, но я рассмотрю только один – алгоритм Грэхема (Graham scan). Этот алгоритм является трехшаговым. На первом шаге ищется любая точка в X, гарантированно входящая в МВО. Обычно выбирается крайняя левая точка (точка с наименьшей x-координатой). Можно начинать построение с крайней правой точки, алгоритма это не меняет. Эту точку называют стартовой, и вся дальнейшая работа будет производиться с оставшимися точками. Второй шаг в алгоритме Грэхема – сортировка оставшихся точек по степени их левизны относительно стартовой точки R. Будем говорить, что B<C, если точка C находится по левую сторону от вектора RB.



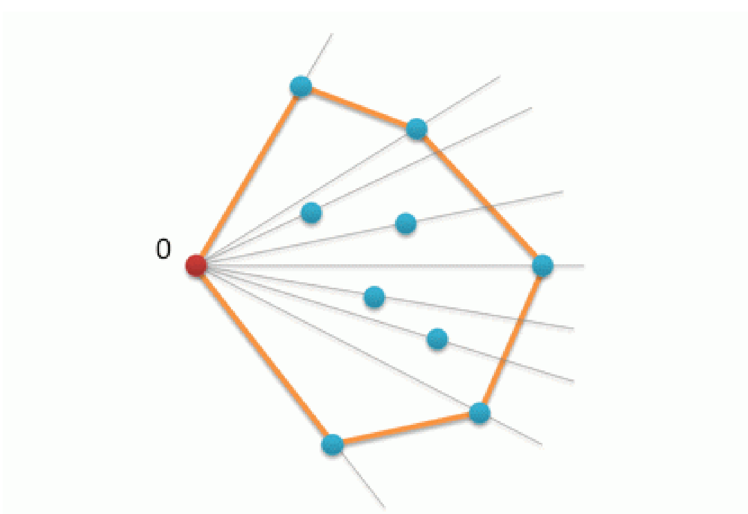
Результат сортировки можно проиллюстрировать следующим рисунком:



Но если соединить все точки в указанном порядке, то получим многоугольник, который, однако, не является выпуклым.



Переходим к третьему шагу алгоритма. Нужно пройтись по всем вершинам и удалить из них те, в которых выполняется правый поворот (угол в такой вершине оказывается больше развернутого). Результатом этого шага будет следующее множество:

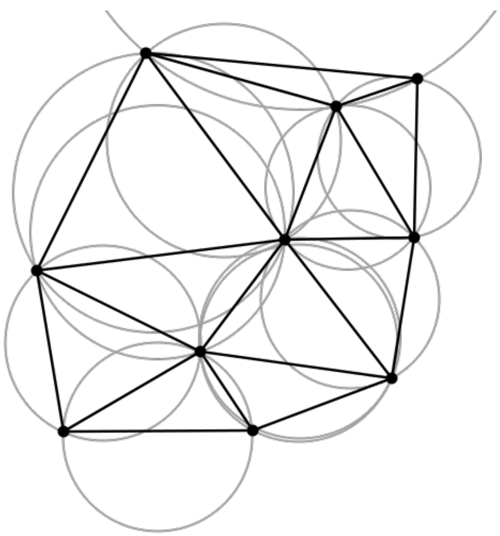


Это и есть минимальная выпуклая оболочка.

3 Применение.

Задача построения выпуклой оболочки является фундаментальной задачей вычислительной геометрии. Важность этой задачи объясняется не только огромным количеством приложений (в распознавании образов, обработке изображений, базах данных, в задаче раскроя и компоновки материалов, математической статистике), но также и полезностью выпуклой оболочки как инструмента решения множества задач вычислительной геометрии. Эта задача позволяет разрешить целый ряд других, иногда с первого взгляда не связанных с ней вопросов: построение диаграмм Вороного, построение триангуляций и т.д. Более подробно рассмотрим построение триангуляций Делоне.

Триангуляция называется выпуклой, если минимальный многоугольник, охватывающий все ее треугольники, является выпуклым. Такой многоугольник называется выпуклой оболочкой данной триангуляции. Триангуляция называется триангуляцией Делоне, если она является выпуклой и удовлетворяет условию Делоне. Условие Делоне заключается в том, что внутрь окружности, описанной вокруг некоторого треугольника, не должна попасть ни одна точка исходного набора.



3.1 Построение триангуляции Делоне.

Начальная триангуляция Делоне строится на основе только тех точек, которые принадлежат выпуклой оболочке. Создаются треугольники, одной из вершин которых является первая точка выпуклой оболочки, а двумя остальными – все возможные пары соседних точек выпуклой оболочки. Далее выполняется проверка выполнения условия Делоне для всех пар треугольников. Пары треугольников, не удовлетворяющие условию, перестраиваются. После того, как первоначальная триангуляция построена, производится последовательное добавление точек, еще не включенных в триангуляцию. Каждая новая точка локализуется, и происходит добавление новых треугольников. После чего выполняется процесс проверки условия Делоне для всех вновь образованных пар треугольников и делаются необходимые перестроения.

3.2 Практическое применение.

Метеорологам при работе над краткосрочным прогнозом погоды необходимо оценивать складывающуюся синоптическую ситуацию. Помощь в этом оказывают карты текущей погоды. Такие карты, собственно, могли бы расчерчиваться по данным объективного анализа. Однако в процедуре объективного анализа имеется ряд задержек, связанных со сбором всех или, по крайней мере, большей части поступающих в реальном времени данных до некоторого срока отсечения, обычно два-три часа после срока наблюдения, их декодированием, выполнением процедур интерполяции разнородных, часто довольно сложных в интерпретации данных и т.п. Однако метеорологам важно иметь экспресс-анализ данных наблюдений как можно раньше, пусть даже не все данные еще поступили. Эта потребность обусловливает необходимость разработки программы для расчета изолиний по данным, расположенным в нерегулярной сети точек. Эту задачу можно решать двумя способами. Во-первых, можно провести интерполяцию в узлы некоторой регулярной сетки и затем расчертить карты по полученным данным. А во-вторых, можно, как это принято в геодезии, построить триангуляцию, то есть планарный граф, все внутренние области которого являются треугольниками, и рассчитать изолинии по полученной триангуляции. Для реализации в графической системе Isograph был избран второй способ.

4 Приложения.

Векторным произведением векторов a и b в пространстве называется вектор $a \times b$, определяемый следующими тремя условиями:

1. Модуль векторного произведения равен произведению модулей перемножаемых векторов на синус угла между ними:
$\mid a \times b \mid = \mid a \mid * \mid b \mid * \sin(a, b)$;
2. Векторное произведение $a \times b$ перпендикулярно и вектору a и вектору b.
3. Упорядоченная тройка векторов a, b, $a \times b$ имеет положительную ориентацию.

Если координаты векторов равны $a(x1, y1, z1)$ и $b(x2, y2, z2)$, то
$$a \times b = \begin{vmatrix} i & j & k \\ x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \end{vmatrix} = \left(\begin{vmatrix} y_1 & z_1 \\ y_2 & z_2 \end{vmatrix} - \begin{vmatrix} x_1 & z_1 \\ x_2 & z_2 \end{vmatrix}, \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} \right)$$

Правый и левый поворот. Совершается движение от точки A до B, затем от B до C. При движении имеет место левый поворот (движение происходит против часовой стрелки), если $AB \times BC > 0$ и правый, если $AB \times BC < 0$.

5 Список источников.

1. http://habrahabr.ru/post/144921
2. http://www.e-olimp.com/articles/25
3. http://method.meteorf.ru/publ/tr/tr346/k_alf.pdf
4. http://www.inf.tsu.ru/library/Publications/2005/17.pdf