Вычислительная геометрия

Гусев Илья

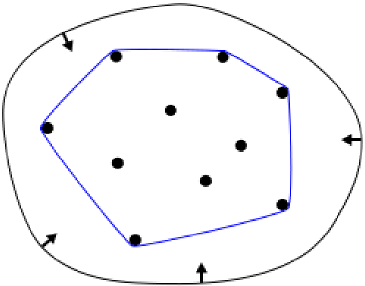
Московский физико-технический институт

Москва, 2017

Содержание

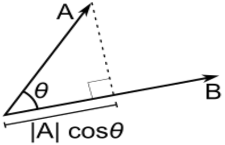
Что это?

1. Рассматриваем конечное множество точек на плоскости (2 координаты).
2. Оболочка множества точек - любая замкнутая кривая без самопересечений такая, что все точки из множества лежат внутри этой кривой.
3. Минимальная оболочка - оболочка минимальной длины (минимального периметра).

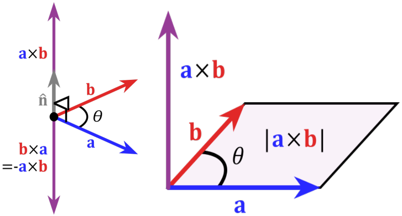


Скалярное и векторное произведение

1. Скалярное произведение:
   $\vec{a} \cdot \vec{b} = a_0 b_0 + a_1 b_1$
   $\vec{a} \cdot \vec{b} = |a||b|cos(\theta)$

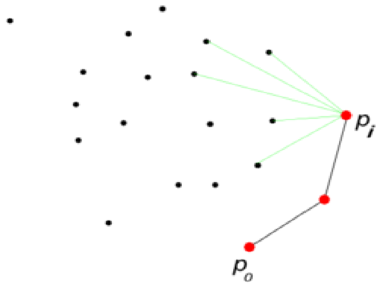


2. Векторное произведение:
   $|\vec{a} \times \vec{b}| = |a_0 b_1 - a_1 b_0$



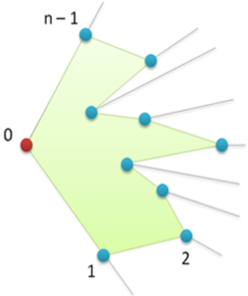
Алгоритм Джарвиса

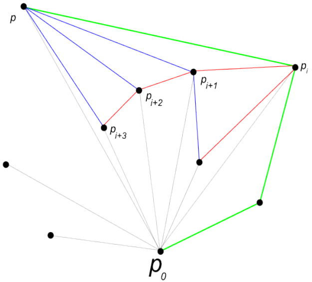
1. Ищем выпуклую оболочку последовательно, против часовой стрелки, начиная с определенной точки.
2. Определённая точка - точно из оболочки, например самая нижняя.
3. На i-ом шаге рассматриваем все оставшиеся точки, и ещё p_0.
4. Ищется косинус угла через скалярное произведение между векторами $p_{i-1}p_i$ и p_ip_{i+1}, где p_{i+1} - претендент на следующую точку оболочки.
5. Выбираем точку с максимальным косинусом (между векторами!).
6. Завершаем, когда снова натыкаемся на p_0. Сложность - $O(n * h)$.



Алгоритм Грэхема

1. Берём самую нижнюю точку (например) и добавляем в ответ.
2. Сортируем все остальные точки по полярному углу относительно p_0.
3. Добавляем в ответ p_1 - самую первую из отсортированных точек.
4. Берем следующую по счету точку t. Пока t и две последних точки в текущей оболочке p_i и p_{i-1} образуют неправый поворот (вектора $p_i t$ и $p_{i-1} p_i$), удаляем из оболочки p_i.
5. Добавляем в оболочку t. Сложность - $O(n * log(n))$.





Алгоритм Эндрю-Грэхема

1. Находим самую левую и самую правую точки множества - p_0 и p_1.
2. Делим множество на две части: точки над и под прямой.
3. Для каждой половины ищем выпуклую оболочку Грехемом с условием, что сортируем не по полярному углу, а по координате.
4. Сливаем получившиеся оболочки.
5. Сложность - $O(n * log(n))$.

Задача про диаметр

Поиск диаметра множества на плоскости за $O(n * log(n))$

1. Строим выпуклую оболочку. (сложность - $O(n * log(n))$)
2. Берём сумму Минковского выпуклой оболочки и минуса выпуклой оболочки ($\vec{a} \to -\vec{a}$). (сложность - $O(n)$)
3. Выбираем максимум по модулю векторов всех вершин суммы Минковского. (сложность - $O(n)$)

Нахождение пары ближайших точек

"Разделяй-и-властвуй"

1. Сортируем точки как пары чисел.
2. Возьмём среднюю после сортировки точку $p_m(m = \lfloor n/2 \rfloor)$, и все точки до неё и саму p_m отнесём к первой половине, а все точки после неё — ко второй половине.
3. $h = \min(h_1, h_2)$, где h1 и h2 - с предыдущего уровня рекурсии, теперь - объединение.
4. $B = \{p_i : |x_i - x_m| < h\}$.
5. $C(p_i) = \{p_j : \ p_j \in B, y_i - h < y_j\}$
6. Стадия объединения: построить B, отсортировать в нём точки по y-координате, затем для каждой точки $p_i \in B$ рассмотреть все точки $p_j \in C(p_i)$, и каждой пары (p_i, p_j) посчитать расстояние и сравнить с текущим наилучшим расстоянием.

Полезные ссылки I

- https://neerc.ifmo.ru: выпуклые оболочки
  https://neerc.ifmo.ru/wiki/index.php?title=Статические_выпуклые_оболочки:_Джарвис,_Грэхем,_Эндрю,_Чен,_QuickHull
- Хабр: выпуклые оболчки
  https://habrahabr.ru/post/144921/
- E-maxx: ближайшие точки
  https://e-maxx.ru/algo/nearest_points