Геометрия. Обзор основых понятий и некоторых алгоритмов

Авторы: Валентин Фондаратов, Тимур Гарипов

Летняя компьютерная школа
Судиславль, "Берендеевы поляны"
Август 2013

1 Базовые определения

1.1 Вектор

- Скалярное произведение $a \cdot b = x_a x_b + y_a y_b$

  $a \cdot b = |a||b| \cos\ \widehat{ab}$

  $a \cdot b = 0 \Leftrightarrow a \perp b$

  Проекция

- Векторное произведение $a \times b = x_a y_b - x_b y_a$

  $a \times b = |a||b| \sin\ \widehat{ab}$

  $a \times b = 0 \Leftrightarrow a || b$

  Ориентированная площадь

- Угол между векторами $\alpha = \mathtt{atan2}(\mathtt{a} \times \mathtt{b}, \mathtt{a} \cdot \mathtt{b})$

  $\mathtt{atan2} \in [-\pi..\pi]$

1.2 Ур-я прямой

1. Аналитическое уравнение $ax + by + c = 0$

   (a, b) — вектор нормали

   По двум точкам: $A = y_2 - y_1, B = x_1 - x_2, C = -ax_1 - by_1$

   Расстояние от точки до прямой $\rho = \frac{|Ax+By+C|}{\sqrt{A^2+B^2}}$

   Нормированное уравнение

2. Параметрическое задание

   $L = \{p \mid \vec{p} = \vec{p_0} + \vec{v}t, t \in \mathbb{R}\}.$

   $\begin{cases} x_p & = & x_{p_0} + x_v t \\ y_p & = & y_{p_0} + y_v t \end{cases}$

2 Пересечения

2.1 2 прямые

$$\begin{cases} A_1 x + B_1 y + C_1 & = & 0 \\ A_2 x + B_2 y + C_2 & = & 0 \end{cases}$$

$$\Delta = \begin{vmatrix} A_1 & B_1 \\ A_2 & B_2 \end{vmatrix}, \Delta_x = \begin{vmatrix} -C_1 & B_1 \\ -C_2 & B_2 \end{vmatrix}, \Delta_y = \begin{vmatrix} A_1 & -C_1 \\ A_2 & -C_2 \end{vmatrix}$$

Либо $\Delta = 0$ и тогда прямые совпадают или не пересекаются, либо

$$\mathbf{x} = \frac{\Delta_x}{\Delta}, \mathbf{y} = \frac{\Delta_y}{\Delta}$$

2.2 Проверка пересечения отрезков

Даны отрезки AB и CD. Пересечение, если $(\overrightarrow{AB} \times \overrightarrow{AC}) \cdot (\overrightarrow{AB} \times \overrightarrow{AD}) \leq 0$ и $(\overrightarrow{CD} \times \overrightarrow{CA}) \cdot (\overrightarrow{CD} \times \overrightarrow{BC}) \leq 0$

Частный случай: $\overrightarrow{AB} \times \overrightarrow{CD} = 0$ — тогда надо проверить пересечение прямоугольников

2.3 Пересечение прямой и окружности

$$\begin{cases} Ax + By + C & = & 0 \\ (x - x_0)^2 + (y - y_0)^2 & = & R^2 \end{cases}$$

Прямая отнормирована. Рисуем картинку, считаем расстояния, прибавляем нормаль с нужным знаком ($Ax_0 + By_0 + C > 0 \Rightarrow -(A, B) \cdot |Ax_0 + By_0 + C|$, иначе $+$. Дальше поворачиваем и получаем точки

2.4 Пересечение двух окружностей

$$\begin{cases} (x - x_0)^2 + (y - y_0)^2 & = & R_0^2 \\ (x - x_1)^2 + (y - y_1)^2 & = & R_1^2 \end{cases}$$

Вычитаем, получаем

$$2x(x_1 - x_0) + 2y(y_1 - y_0) = R_0^2 - R_1^2 + y_1^2 - y_0^2 + x_1^2 - x_0^2$$

Остается пересечь окружность и прямую. Радуемся.

2.5 Касательная к окружности

Находим расстояние до точки касания, строим окружность, пересекаем, радуемся.

2.6 Общая касательная двух окружностей

Внешние касательные: вычитаем меньший из радиусов из обоих, строим касательные из точки, возвращаем на место сдвигом по нормали;

Внутренние касательные: меньший из радиусов прибавляем к большему; меньший сжимаем.

3 Прочие полезности

3.1 Площадь многоугольника

Ориентированная площадь из любой точки плоскости: $Sum = \sum_{i=0}^{n-1} \overrightarrow{(x_i, y_i)} \times \overrightarrow{(x_{i+1}, y_{i+1})}$

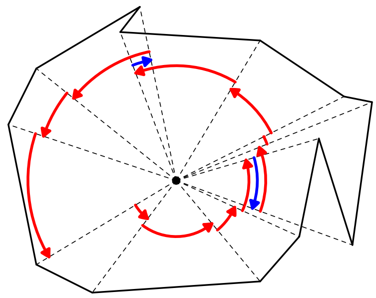
Площадь многоугольника $\left|\frac{Sum}{2}\right|$.

3.2 Проверка на принадлежность точки многоугольнику

Выпуклый многоугольник: сравниваем сумму неориентированных площадей с настоящей

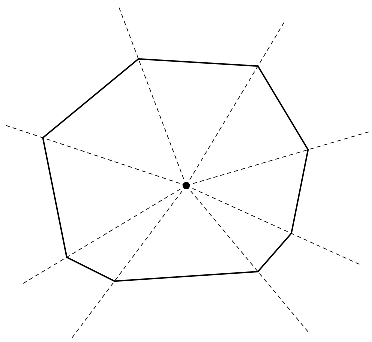
Невыпуклый многоугольник:

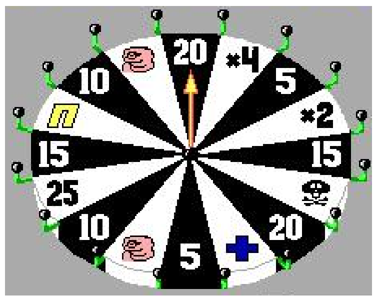
Сумма углов (ориентированных) должна быть примерно 2π



Луч: пускаем случайный/горизонтальный луч. Если горизонтальный, используем полуоткрытые отрезки. Проверяем, что количество пересечений с многоугольником нечетно.

Online за $O(\log n)$: предподсчет за $O(n)$: создаем массив лучей из точки внутри многоугольника, нам дают точку, находим нужный сектор "приз" на барабане с помощью бинпоиска.





4 Более сложные алгоритмы

4.1 Две ближайшие точки за $\mathrm{O}\big(n\log n\big)$

Алгоритм устроен по принципу «разделяй и властвуй».

Алгоритм 1: Нахождение двух ближайших точек

Вход: n точек: $x_i \leq x_{i+1}, i < n$

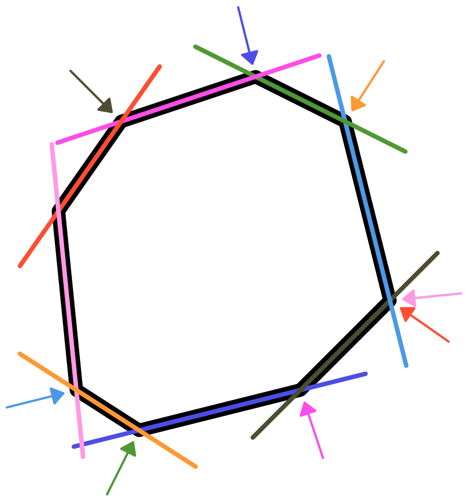
Выход: L — кратчайшее расстояние между точками; сортировка точек $p_i : y_i \leq y_{i+1}$

if $n = 2$ **then**
    return $\rho(p_0, p_1)$
end
if $n = 1$ **then**
    return ∞
end
$(L^{(1)}, p^{(1)}) = nearest(p_0, \ldots, p_{n/2-1})$;
$(L^{(2)}, p^{(2)}) = nearest(p_{n/2+1}, \ldots, p_{n-1})$;
$L = \min(L^{(1)}, L^{(2)})$;
$p^{ans} = merge(p^{(1)}, p^{(2)})$;
$p^{lane} = p^{ans} \bigcap [x_{n/2} - L, x_{n/2} + L]$;
for $i = 0 \to |p^{lane}| - 1$ **do**
    for $j = i + 1 \to \min(i + 6, |p^{lane}| - 1)$ **do**
        if $dist(p_i^{lane}, p_j^{lane}) < L$ **then**
            $L = dist(p_i^{lane}, p_j^{lane})$;
        end
    end
end
return (L, p^{ans})

4.2 Две наиболее удаленные точки за $\mathrm{O}\big(n\log n\big)$

1. Строим выпуклую оболочку точек;

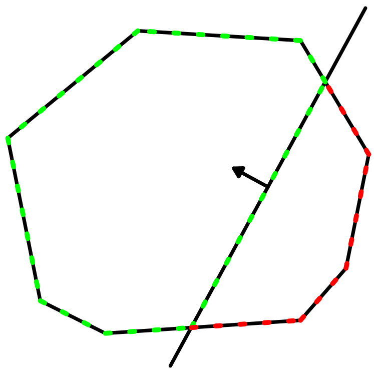
2. Для каждой стороны ищем самую дальнюю точку от содержащей ее прямой и релаксируем ответ расстоянием от нее до прилегающей к рассматриваемой стороне вершины.



Поиск наидальнейшей точки производится с помощью метода двух указателей.

4.3 Пересечение полуплоскостей за $O(n^2)$

На k-ом шаге алгоритма храним выпуклый многоугольник, являющийся пересечением первых k полуплоскостей. При добавлении очередной полуплоскости необходимо исключить, те вершины которые в ней не лежат, и, возможно, добавить новые вершины на пересечении многоугольника и прямой, ограничивающей полуплоскость. Вершины многоугольника удобно хранить в массиве или списке.



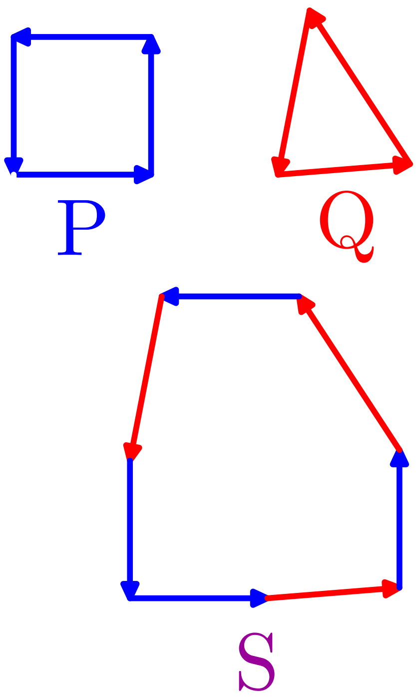
4.4 Сумма Минковского за $O(n)$

Определение: Суммой Минковского двух множеств точек P и Q называют множество $S = P + Q = \{s \mid \vec{s} = \vec{p} + \vec{q},\ p \in P,\ q \in Q\}$.

Полезные факты:

1. Если P и Q — выпуклые множества, то множество $P + Q$ тоже выпуклое.

2. Обозначим множетсво точек, являющееся границей выпуклого множества A, как $\Gamma(A)$. Пусть P и Q — выпуклые множества, тогда $\Gamma(P+Q) \subset \Gamma(P) + \Gamma(Q)$.
3. Сумма Минковского выпуклых многоугольников — выпуклый многоугольник.
4. Чтобы получить сумму Минковского двух выпуклых многоугольников, достаточно представить многоугльники как массивы векторов-сторон отсортированых по полярному углу, и слить их в один отсортированный массив. Полученный массив содержит в себе стороны суммы Минковского в порядке обхода.



4.5 Проверка на пересечение выпуклых многоугольников за $O(n)$

Рассмотрим выпуклые многоугольники P и Q, построим многоугольник $-Q = \{z \mid -\vec{z} \in Q\}$. P и Q пересекаются тогда и только тогда, когда $(0,0) \in P + (-Q)$.