

半平面交的算法及其应用

北京四中
李澎煦

基本概念

- **半平面**：平面上的直线及其一侧的部分。
- 半平面可由不等式 $ax+by+c \geq 0$ 确定。
- 在一个有界区域里半平面或半平面的交是一个凸多边形区域。
- n 个半平面的交是一个至多 n 条边的凸多边形。

半平面交的联机算法

procedure intersection of half-planes

输入: n 个半平面 H_1, H_2, \dots, H_n

输出: $H_1 \cap H_2 \cap \dots \cap H_n$

初始化区域 A 为整个平面

依次用直线 $a_i x + b_i y + c_i = 0$, $i = 1, 2, \dots, n$ 切割

A , 保留使不等式 $a_i x + b_i y + c_i \geq 0$ 成立的部分

输出 A

复杂度 $O(n^2)$, 联机算法。

半平面交的分治算法

假设可以在 $O(m+n)$ 的时间内将 m 个半平面的交和 n 个半平面的交合并，则可以有一种 $O(n*\log(n))$ 的分治算法求半平面的交。

Procedure intersection of half-plane (D&C)

输入: n 个半平面 H_1, H_2, \dots, H_n

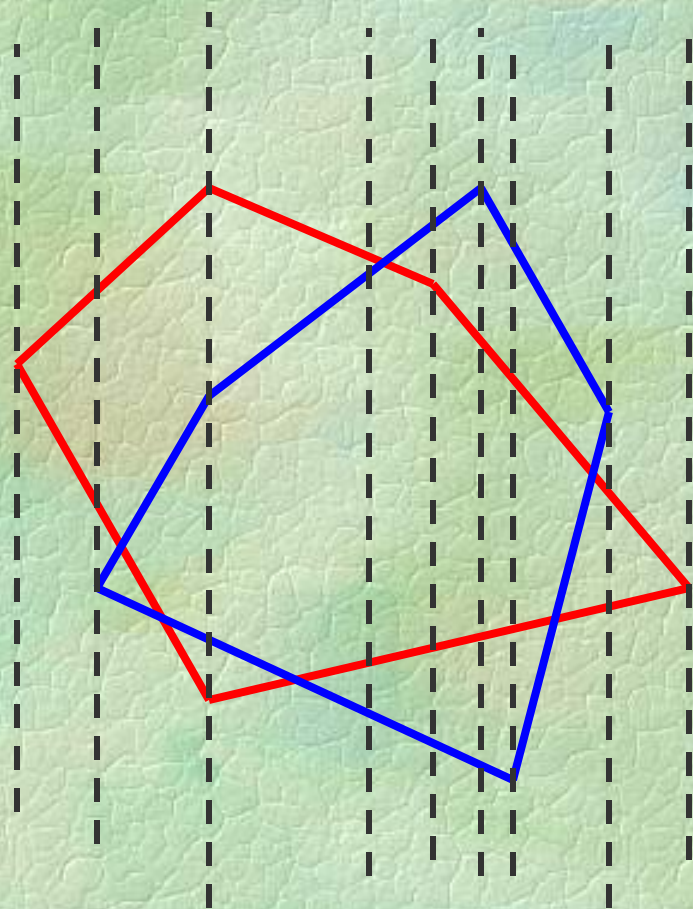
输出: $H_1 \cap H_2 \cap \dots \cap H_n$

将 $H_1 \dots H_n$ 分成两个大小近似相等的集合

在每个子问题中递归地计算半平面的交

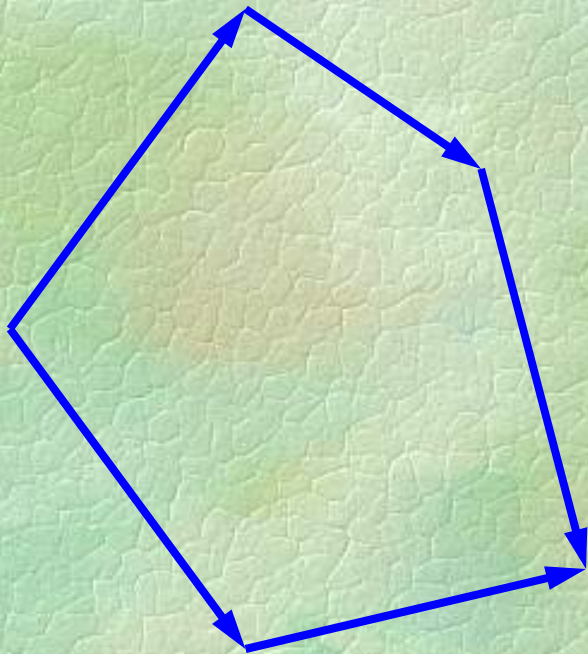
合并两个凸多边形区域形成 $H_1 \cap H_2 \cap \dots \cap H_n$

问题的关键是怎样在 $O(m+n)$ 的时间里求两个凸多边形的交。



- 将两个凸多边形沿顶点切割成至多 $O(m+n)$ 个平行于 y 轴的梯形区域
- 每两个梯形区域的交可以在 $O(1)$ 时间内解决

描述凸多边形的方法



- 凸多边形上方和下方的顶点分别构成一个 x 坐标递增序列。
- 将这两个序列中的顶点分别作为一个链表存储，得到确定凸多边形区域的上界和下界。

凸多边形交的算法 1 :

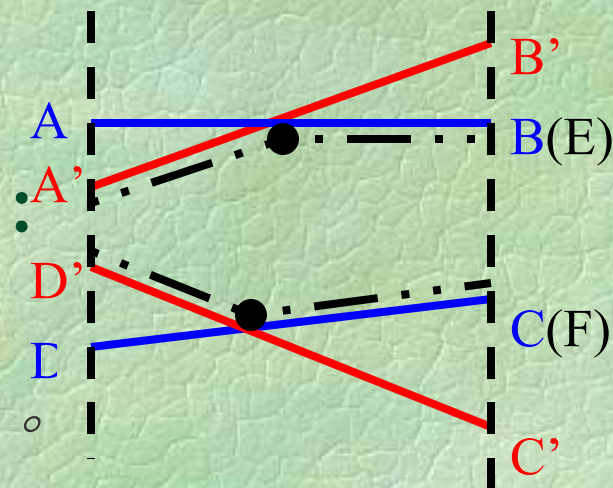
procedure intersection of convex polygon

输入：两个凸多边形区域 A 、 B

输出： $C=A\cap B$

1. 将两个凸多边形的顶点 x 坐标分类，得到序列 $x_i, i=1 \dots p$
2. 初始化区域 C 为空。
3. 处理 $\{x_i\}$
4. 依次处理区域 $(x_i, x_{i+1}], i=1 \dots p-1$ 。
5. 输出 C

凸多边形交的算法 2



4. 依次处理区域 $(x_i, x_{i+1}]$, $i=1 \dots p-1$ 。

4.1 计算两个多边形在此区域里截得的梯形（可能退化）： $ABCD$ 和 $A'B'C'D'$ 。

4.2 求交点 $AB \cap A'B'$ 、 $AB \cap C'D'$ 、 $CD \cap A'B'$ ，将存在的点按 x 坐标排序，删除重复，添加到 C 的上界中。

4.3 用类似的方法求 C 的下界

4.4 计算此区域的右侧边界（线段的交）：

$EF = BC \cap B'C'$ 。将 E 、 F 分别加入到 C 的上界和下界中。

算法的复杂度

- 步 1：由于 A 、 B 的上下界 x 坐标分别有序，可采用归并排序。复杂度 $O(m+n)$
- 步 4：由于是按照 x 递增的顺序扫描这些区域，每条边界上的指针在整个过程中始终向右移动。两个多边形的每个顶点至多扫描一次。复杂度为 $O(m+n)$ 。
- 整个算法的时间复杂度为 $O(m+n)$ 。

问题 1

Hotter and Colder

(Waterloo local contest)

A 和 B 在 $10*10$ 的棋盘上进行一个游戏。 A 确定一个点 P ， B 每回合移动一次。每次 A 都会告诉 B ，他当前所处的位置是离 P 更近了 (*Hot*) 还是更远了 (*Cold*)。

(原题还要考虑距离不变的情况。)

请在 A 每次回答后，确定 P 点可能存在的区域的面积。

问题 1 分析：

- 假设 B 从 $C(x_1, y_1)$ 移动到了 $D(x_2, y_2)$ ， A 回答 *Hot*。那么 $P(x, y)$ 所处的位置就满足 $|CP| > |DP|$ ，即：

$$2(x_2 - x_1)x + 2(y_2 - y_1)y + x_1^2 + y_1^2 - x_2^2 - y_2^2 > 0$$

- 类似地，回答 *Cold* 对应于另一个不等式。
- 初始时可能的区域是 $[0, 10] * [0, 10]$ 。每回合后都用相应的不等式对应的半平面与当前区域求交。并输出交的面积。

问题 2

Nice Milk (OOPC1)

SRbGa 有一块凸 n 边形面包，和一盆面积足够大但深度仅为 h 的牛奶。他想仅蘸 k 次（每次都保证面包垂直于盆底），使得面包蘸上牛奶的部分面积最大。

问题 2 分析

:

- 由于本题规模不大，考虑使用深度优先搜索。
- 蘸每条边都对应剩下的一个半平面，某种蘸 k 条边 $E_1 \dots E_k$ 的方法，剩下的部分就对应于这 k 个半平面和原多边形的交。
- 考察 $C(n, k)$ 种蘸法，选其中剩下面积最小的那种。

小结

问题 1 是用几个半平面顺次求交，并且每次都要输出面积。显然采用联机算法合适。

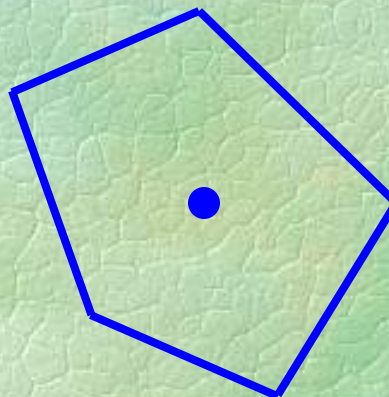
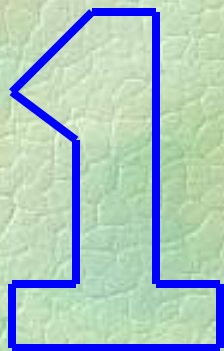
问题 2 如果用联机算法，复杂度为 $O[C(n, k) * \underline{n}]$ ，且便于在搜索的过程中剪枝。如果用脱机的分治算法，复杂度为 $O[C(n, k) * \underline{(n + k * \log(k))}]$ 。

问题

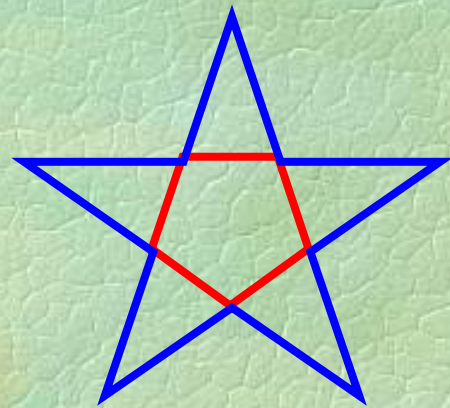
3 :

Video (CTSC98)

已知一个多边形 P （不一定是凸的）问在 P 中是否存在点 Q ，在 Q 点能观察到整个多边形区域。



问题 3 分析：



- 若多边形的顶点按逆时针顺序给出 $V_0V_1V_2\ldots V_n$ ， $V_0=V_n$ 。则能够观察到边 V_iV_{i+1} 的点 Q_i 一定满足 $\overrightarrow{Q_iV_i} * \overrightarrow{Q_iV_j} \geq 0, i=0\ldots n-1$
- 能观察到所有边的点一定能够观察到整个多边形区域。
- 如果用坐标进行叉积运算，则每个约束条件都对应一个二元一次不等式（也对应于一个半平面）。本题就转化为求这 n 个半平面的交是否不为空。

问题

4 :

Triathlon (NEERC2000)

n 名选手参加铁人三项赛，比赛按照选手在三个赛段中所用的总时间排定名次。已知每名选手在三个项目中的速度 U_i 、 V_i 、 W_i 。

问对于选手 i ，能否通过适当的安排三个赛段的长度（但每个赛段的长度都不能为 0），来保证他获胜。

问题 4 分析

:

- 假设三个赛段的长度分别为 x 、 y 、 z ，则选手 i 获胜的充要条件就是：

$$\frac{x}{u_i} + \frac{y}{v_i} + \frac{z}{w_i} < \frac{x}{u_j} + \frac{y}{v_j} + \frac{z}{w_j}$$

- 这是一个三元齐次不等式组，由于 $z > 0$ ，所以不妨将每个不等式两侧都除以 z ，并令 $X = x/z$ ， $Y = y/z$ ，就得到

$$\left(\frac{1}{u_j} - \frac{1}{u_i} \right) X + \left(\frac{1}{v_j} - \frac{1}{v_i} \right) Y + \left(\frac{1}{w_j} - \frac{1}{w_i} \right) > 0$$

- 本题就转化为求这 $n-1$ 个不等式对应的半平面的交，并判断其面积是否大于 0（即排除空集、点、线的情况）。

小结

问题 3 和问题 4，最终都转化为二元不等式组解的存在性问题。可以用半平面交的分治算法有效地解决。但两个问题又略有不同，一个是 $=0$ 、一个是 ≥ 0 。也就是说对多边形的边界处理不同。 ≥ 0 的不等式要考虑退化为点、线的情况，稍微复杂一点。

问题

5 :

Run away (CERC99)

在一个矩形 R 中有 n 个点 $P_1...P_n$ ，请找出一个点 $Q \in R$ 使 $\min(|QP_i|)$ 最大。

问题 5 分析 1 :

- 将 R 分成 n 个区域, $Q_1 \dots Q_n$, Q_i 是 R 里离 P_i 点的距离比离其它点都小的点的集合:

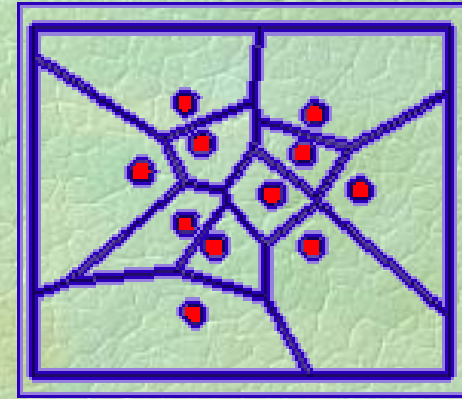
$$Q_i = \left\{ Q \mid |QP_i| \leq |QP_j|, i \neq j \right\} \cap R$$

- Q_i 可通过在 $P_i P_j$ 的中垂线 P_i 一侧的半平面的交求得。
 Q_i 为一个凸多边形。
- 在 Q_i 里, 离 P_i 最远的点只能出现在 Q_i 的顶点上。求其中最远的点即可。

问题 5 分析 2 :

- 半平面的交采用分治算法, 每个点的复杂度为 $O(n*\log(n))$ 。
- 对应于 P_i 的多边形最多有 $O(n)$ 个顶点, 因此求 Q_i 中的最远点复杂度为 $O(n)$ 。
- 总的复杂度为 $O(n*n*\log(n))$ 。

Voronoi 图



- 实际上，由以上方法定义的 n 个多边形区域 $Q_1 \dots Q_n$ 就组成了一个 **Voronoi 图**。
- *Voronoi* 图是计算几何中仅次于凸包的几何对象，有着非常广泛的应用。
- 利用半平面的交求 *Voronoi* 图的算法不是最优的。
- 分治法、平面扫描法等许多算法都能达到 $O(n \cdot \log(n))$ 的复杂度，这才是最优的。但这些算法都过于复杂，不属于本文讨论的范围。

半平面交的算法及其应用

基本概念

算法

半平面交的联机算法

半平面交的分治算法

应用

问题 1 : **Hotter and Colder**

问题 2 : **Milk**

小结

问题 3 : **Video**

问题 4 : **Triathlon**

小结

问题 5 : **Run away**

Voronoi 图