

物理实验 (laser) 解题报告

quailty

2018 年 5 月 15 日

物理实验 (laser.c/cpp/pas)

- 二维平面上有一条直线，直线上放置了一个激光发射器，会向导轨两侧沿导轨垂直方向发射宽度为 L 的激光束。平面上还有 n 条线段，并且线段和线段、线段和直线之间都没有公共点，线段和直线的夹角不超过 85° ，激光束不能穿透线段，你需要求出激光束能照射到的线段长度之和的最大值。要求相对误差不超过 10^{-6} 。

物理实验 (laser.c/cpp/pas)

- 二维平面上有一条直线，直线上放置了一个激光发射器，会向导轨两侧沿导轨垂直方向发射宽度为 L 的激光束。平面上还有 n 条线段，并且线段和线段、线段和直线之间都没有公共点，线段和直线的夹角不超过 85° ，激光束不能穿透线段，你需要求出激光束能照射到的线段长度之和的最大值。要求相对误差不超过 10^{-6} 。
- 对于 100% 的数据， $T \leq 20$ ， $2 \leq n \leq 2 \times 10^5$ ， $1 \leq L \leq 10^9$ ，所有坐标的绝对值不超过 10^9 。

物理实验 (laser.c/cpp/pas)

- 二维平面上有一条直线，直线上放置了一个激光发射器，会向导轨两侧沿导轨垂直方向发射宽度为 L 的激光束。平面上还有 n 条线段，并且线段和线段、线段和直线之间都没有公共点，线段和直线的夹角不超过 85° ，激光束不能穿透线段，你需要求出激光束能照射到的线段长度之和的最大值。要求相对误差不超过 10^{-6} 。
- 对于 100% 的数据， $T \leq 20$ ， $2 \leq n \leq 2 \times 10^5$ ， $1 \leq L \leq 10^9$ ，所有坐标的绝对值不超过 10^9 。
- Shortest judge solution: 2888 bytes

40 分做法

- 对于第 1 个测试点, $n \leq 2000$, 所有坐标的绝对值不超过 10^4 。

40 分做法

- 对于第 1 个测试点, $n \leq 2000$, 所有坐标的绝对值不超过 10^4 。
- 过每个线段的端点向直线作垂线, 将直线分割成若干个区间, 同时将每条线段也分割成若干段。

40 分做法

- 对于第 1 个测试点, $n \leq 2000$, 所有坐标的绝对值不超过 10^4 。
- 过每个线段的端点向直线作垂线, 将直线分割成若干个区间, 同时将每条线段也分割成若干段。
- 对每个区间考虑覆盖单位长度的贡献, 分别求出直线两侧的区间内的最近线段, 根据线段与直线的夹角计算覆盖单位长度的贡献。

40 分做法

- 假设已经求出了每个区间内覆盖单位长度的贡献，现在需要在直线上放一个长度为 L 的线段使得贡献之和最大。

40 分做法

- 假设已经求出了每个区间内覆盖单位长度的贡献，现在需要在直线上放一个长度为 L 的线段使得贡献之和最大。
- 不难发现线段一定会卡在某个区间的端点上，否则左右抖动这个线段不会使答案变差。

40 分做法

- 假设已经求出了每个区间内覆盖单位长度的贡献，现在需要在直线上放一个长度为 L 的线段使得贡献之和最大。
- 不难发现线段一定会卡在某个区间的端点上，否则左右抖动这个线段不会使答案变差。
- 枚举线段的左/右端点卡住的端点，计算此时的贡献更新答案即可。

40 分做法

- 假设已经求出了每个区间内覆盖单位长度的贡献，现在需要在直线上放一个长度为 L 的线段使得贡献之和最大。
- 不难发现线段一定会卡在某个区间的端点上，否则左右抖动这个线段不会使答案变差。
- 枚举线段的左/右端点卡住的端点，计算此时的贡献更新答案即可。
- 复杂度 $O(n^2)$ ，期望得分 40 分。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。
- 两部分算法都需要优化。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。
- 两部分算法都需要优化。
- 为了方便讨论, 先将坐标系平移、旋转使得给定直线是 x 轴。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。
- 两部分算法都需要优化。
- 为了方便讨论, 先将坐标系平移、旋转使得给定直线是 x 轴。
- 沿 x 轴正方向进行扫描线, 用两个平衡树分别按照从近到远的顺序维护 x 轴两侧与扫描线有交的线段。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。
- 两部分算法都需要优化。
- 为了方便讨论, 先将坐标系平移、旋转使得给定直线是 x 轴。
- 沿 x 轴正方向进行扫描线, 用两个平衡树分别按照从近到远的顺序维护 x 轴两侧与扫描线有交的线段。
- 在相邻两个事件之间查询两侧最近的线段, 根据和 x 轴的夹角计算当前区间覆盖单位长度的贡献。

80 分做法

- 对于第 2 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^4 。
- 两部分算法都需要优化。
- 为了方便讨论, 先将坐标系平移、旋转使得给定直线是 x 轴。
- 沿 x 轴正方向进行扫描线, 用两个平衡树分别按照从近到远的顺序维护 x 轴两侧与扫描线有交的线段。
- 在相邻两个事件之间查询两侧最近的线段, 根据和 x 轴的夹角计算当前区间覆盖单位长度的贡献。
- 实际上只需要 `std::set`。

80 分做法

- 求出每个区间内覆盖单位长度的贡献之后，从左往右枚举线段右端点卡住的端点。

80 分做法

- 求出每个区间内覆盖单位长度的贡献之后，从左往右枚举线段右端点卡住的端点。
- 线段左端点所在的区间会随着右端点不断右移，可以直接维护，计算此时的贡献更新答案即可。

80 分做法

- 求出每个区间内覆盖单位长度的贡献之后，从左往右枚举线段右端点卡住的端点。
- 线段左端点所在的区间会随着右端点不断右移，可以直接维护，计算此时的贡献更新答案即可。
- 别忘了反过来再做一次。

80 分做法

- 求出每个区间内覆盖单位长度的贡献之后，从左往右枚举线段右端点卡住的端点。
- 线段左端点所在的区间会随着右端点不断右移，可以直接维护，计算此时的贡献更新答案即可。
- 别忘了反过来再做一次。
- 复杂度 $O(n \log n)$ ，期望得分 80 分。

100 分做法

- 对于第 3 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^9 。

100 分做法

- 对于第 3 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^9 。
- 需要优化精度。

100 分做法

- 对于第 3 个测试点, $n \leq 10^4$, 所有坐标的绝对值不超过 10^9 。
- 需要优化精度。
- 如果直接用 double 或 long double 冲过去了, 这 20 分就当送温暖了。

100 分做法

- 对于扫描线部分，可以做到全整数。

100 分做法

- 对于扫描线部分，可以做到全整数。
- 平移、旋转坐标系是不必要的，只需要将事件点按照与直线方向向量的点积从小到大排序即可实现沿直线方向的扫描线。

100 分做法

- 对于扫描线部分，可以做到全整数。
- 平移、旋转坐标系是不必要的，只需要将事件点按照与直线方向向量的点积从小到大排序即可实现沿直线方向的扫描线。
- 判断线段在直线的哪一侧只需要任取一个端点判断与直线方向向量的叉积的符号。

100 分做法

- 对于扫描线部分，可以做到全整数。
- 平移、旋转坐标系是不必要的，只需要将事件点按照与直线方向向量的点积从小到大排序即可实现沿直线方向的扫描线。
- 判断线段在直线的哪一侧只需要任取一个端点判断与直线方向向量的叉积的符号。
- 插入/删除线段的时候比较当前事件点和与扫描线有交的线段的相对位置，每次比较只需要判断事件点在线段上方还是下方，仍然是叉积。

100 分做法

- 对于扫描线部分，可以做到全整数。
- 平移、旋转坐标系是不必要的，只需要将事件点按照与直线方向向量的点积从小到大排序即可实现沿直线方向的扫描线。
- 判断线段在直线的哪一侧只需要任取一个端点判断与直线方向向量的叉积的符号。
- 插入/删除线段的时候比较当前事件点和与扫描线有交的线段的相对位置，每次比较只需要判断事件点在线段上方还是下方，仍然是叉积。
- long long 就够了。

100 分做法

- 对于双指针部分，没有优化的必要。

100 分做法

- 对于双指针部分，没有优化的必要。
- 复杂度 $O(n \log n)$ ，期望得分 100 分。

Thank you!