

作业3 实验报告

刘喆骐 2020013163 探微化01

1. 实验环境

Windows, vscode, C++, mingw, QT (版本5.9.9)。

2. 算法

2.1 暴力法

通过遍历所有的点，计算两两之间的距离，进而判断出最短距离。由于不需要重复计算两点间的距离，取第 k 个点时只需计算其与剩余 $n-k$ 个点的距离，故一共需要计算 $\sum_{k=1}^n k(n-k)$ 次，故时间复杂度为 $\Theta(n^2)$ 。

2.2 分治法

分治法首先将所有的点按照 x 轴进行排序，得到有序数列 $[left, right]$ ，这步操作的时间复杂度为 $O(n \log n)$ 。然后进行分治，将 $[left, right]$ 分成 $[left, mid]$, $[mid, right]$ ，分别得出最小距离，并取二者的最小值 min_d 。此时还需考虑 $(mid - min_d, mid + min_d)$ 之间的点，将这些点按照 y 坐标的大小降序排序。对于 $i < j$ ，找出 $y_i - y_j < min_d$ 的点，然后比较 $distance(i, j)$ 与 min_d ，由于鸽巢原理，符合条件的区域内的点仅有6个。于是，分治法时间复杂度满足如下关系：

$$T(n) = T\left(\frac{n}{2}\right) + O(n)$$

由主定理可知，分治的时间复杂度为 $O(n \log n)$ ，故总的时间复杂度为 $O(n \log n) + O(n \log n) = O(n \log n)$ 。

3. 实验过程

在 $n=5, 10, 100, 1000, 10000$ 下对于 $O(n^2)$, $O(n \log n)$ 算法进行耗时比较，然后对于 $n=100000, 200000, \dots, 1000000$ ，观察 $O(n \log n)$ 算法的耗时。使用的排序方法为mergesort。

4. 结果和分析

结果如下图所示，其中时间的单位为ms。

n	暴力法耗时	分治法耗时
5	0	0
10	0	0
100	0	0
1000	13	1
5000	332	5
10000	1312	10
100000	–	126
200000	–	257
300000	–	422
400000	–	562
500000	–	781
600000	–	871
700000	–	998
800000	–	1141
900000	–	1369
1000000	–	1591

图1 原始数据

观察发现，暴力法在 $n > 10000$ 时耗时已经超过1s，而分治法耗时则很少，故暴力法是一种低效的算法。对于暴力法的数据进行拟合，得到下图。

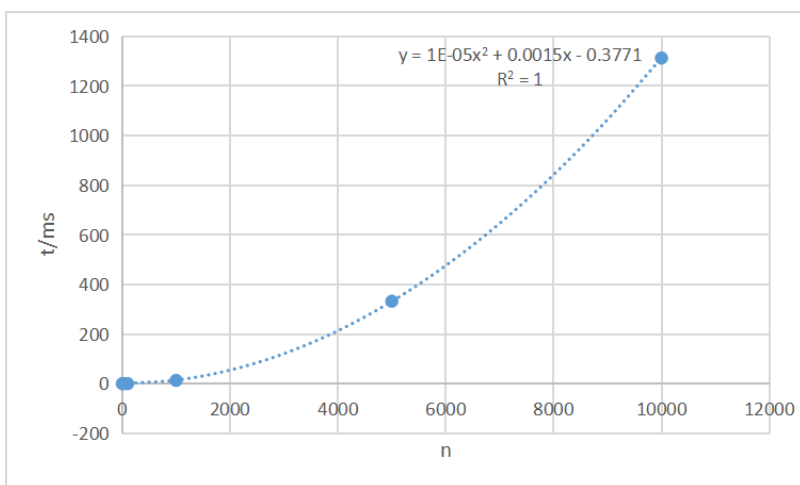


图2 暴力法时间-n关系

可以看到， t - n 满足较好的二次函数的关系， $t = 10^{-5}n^2 + 0.0015n - 0.3771$ ， $R^2 = 1$ 。符合暴力法时间复杂度为 $O(n^2)$ 。

对于分治法进行拟合，得到下图：

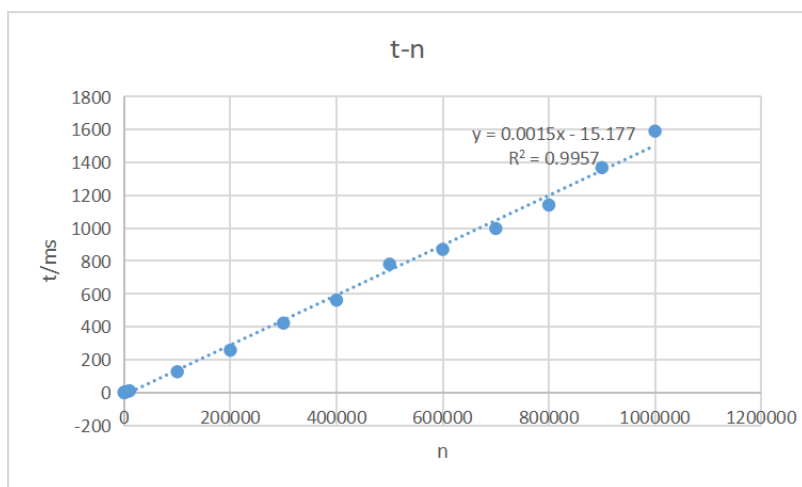


图3 分治法时间-n关系

发现在1000000的量级下，时间和n成接近线性的关系，而且有波动，使用线性拟合得到 $y = 0.0015n - 15.177$, $R^2 = 0.9957$ 。于是再拟合t-nlogn关系，得到下图：

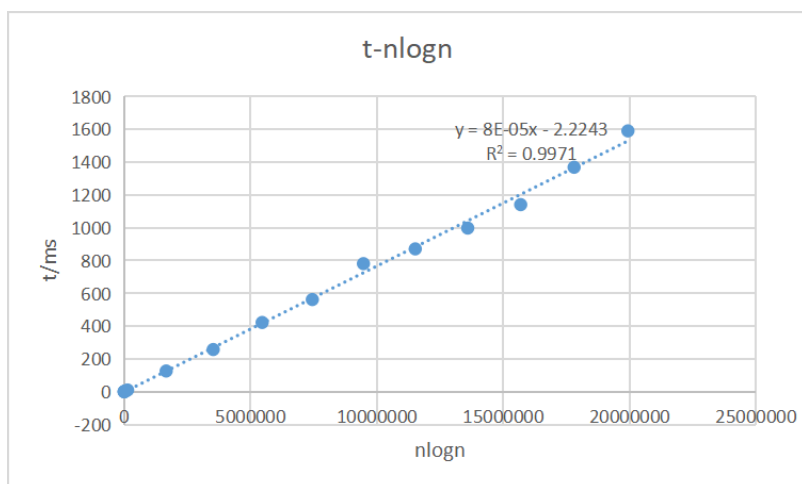


图4 分治法时间-nlogn关系

发现时间和nlogn成接近线性的关系，而且依旧有波动， $y = 8 * 10^{-5}n \log n - 2.2243$, $R^2 = 0.9971$ 。结果较为符合预期。

5. 总结

本问题使用分治法会得到比暴力求解好的多的耗时。但是还有非分治法和期望线性的做法，留待以后学习。^[1]

6. 参考文献

[1]<https://oi-wiki.org/geometry/nearest-points/#非分治算法>