

平面最近点对 实验报告

吴佳龙 2018013418

摘要

本次实验结合理论分析和程序设计，运用了暴力算法和分治算法分别计算平面最近点对，并验证了它们的结果正确性，比较了它们的计算时间，实验结果与理论分析相符。此外，还实现了带有图形界面的演示程序，能够通过鼠标和键盘交互输入平面点，并实时标识最近点对。

1 问题

实现求平面上最近点对的 $\Theta(n \lg n)$ 算法，分析比较在不同输入规模下 $\Theta(n^2)$ 和 $\Theta(n \lg n)$ 算法的实际运行时间。

```
pair<int, int> closest_divide_and_conquer(  
    const Point * p, int n);
```

2 实验环境

操作系统: Windows 10
IDE: Visual Studio 2017
处理器: 3.1 GHz 双核 Intel Core i5
图形界面开发环境: Qt 5.12.4 with Qt Creator 4.9.1

算法原理 将 n 个点通过一条竖直平分线均匀地分至平分线的两侧，并分别求解规模为 $\frac{n}{2}$ 的子问题，记其最近点对距离分别为 d_1, d_2 ，令 d 为两者中较小者。

若存在一对点，他们分别位于平分线的两侧，但是他们的距离小于 d ，那么这些点一定位于以平分线为中心，左右各 d 的带状区域中。取出该带状区域中所有点并按照 y 坐标排序，可以证明若存在一对点距离小于 d ，他们排序后的位置一定相差不超过 7。具体证明可见《算法导论》33.4 节。

3 算法分析

本次实验共两种算法，分别在以下 2 个小节中进行算法描述与分析。

3.1 暴力算法

```
pair<int, int> closest_brute(const Point * p  
    , int n);
```

算法描述 枚举所有共 $\binom{n}{2}$ 对点对，计算距离并取最小值。

时间复杂度分析 $\Theta(n^2)$

3.2 分治算法

时间复杂度分析 为了使排序的复杂度不影响到分治算法的总复杂度，我们需要在递归调用分治算法前进行预排序，将所有点按照 x 坐标和 y 坐标分别排序并将排序后的序号存入数组 X, Y 传递进递归函数。在分治的过程中，将当前所有点分为均匀的两个子问题，由于已经存在排好序的 X, Y ，将其分裂为左右两部分各自的 X, Y 仅需 $\Theta(n)$ 次操作。解递归式

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

得

$$T(n) = \Theta(n \lg n)$$

使用基于比较的 $\Theta(n \lg n)$ 的排序算法，算法的总复杂度（包括排序）为

$$\Theta(n \lg n) + T(n) = \Theta(n \lg n)$$

4 结果分析

4.1 结果正确性

首先我们验证了算法实现的正确性，验证方法为：随机生成 n 个平面上的点，并调用不同方法计算他们的乘积，再比较暴力算法和分治算法的结果。注意这里我们仅比较最近点对的距离是否一致，因为最近点对可能不唯一。

实验表明，对于 n 个平面点，暴力算法和分治算法的计算结果总是相同的。由此可以认为我们对于算法的实现是无误的。

4.2 计算时间

在不同尺度的输入规模 n 的情况下，两算法计算一次最近点对所需的时间如图 1, 2, 3。

结果分析 暴力算法的运行时间增长呈一条下凸曲线，在 $n \approx 20000$ 时，运行时间接近 1 秒，这与其 $\Theta(n^2)$ 的时间复杂度是相符合的。

分治算法的运行时间远远少于暴力算法，符合其 $\Theta(n \lg n)$ 的复杂度。当 $n \in [10^5, 2 \times 10^6]$ 时，其运行时间接近线性增长，可能的原因是 n 还不够大，不足以显现其增长的非线性。

5 总结：不同方法的比较

暴力方法 时间复杂度为 $\Theta(n^2)$ ，实现及其简单。

分治算法 时间复杂度为 $\Theta(n \lg n)$ ，特别巧妙地运用了欧式空间中点的距离度量的三角不等式，可能可以扩展到更多度量空间。比如三维欧式空间：可以证明，在 \mathbb{R}^3 中，在合并两个子问题的过程中，仅需检查带状区域中一个点与其后的 15 个点。

A 图形界面程序说明

图形界面程序 `Geometry_GUI_boxed.exe` 运用 Qt 开发，能够交互地通过鼠标点击或文本输入平面点，并实时标识最近点对及其距离。点击即可运行，具体功能如下：

1. 界面左侧为画布，起始绘制两条坐标轴，左下角状态栏实时显示鼠标所在坐标，单击鼠标可插入点。
2. 右侧为当前所有点的列表，右下方也可通过键盘输入坐标来插入点。
3. 右键点击点列表中的点可删除相应的点。
4. 画布实时通过醒目的颜色标识最近点对，点列表下方亦实时显示最近点对的坐标和距离。
5. 界面上方工具栏有三个按钮，分别为：
 - Fit Window：将原点置于画布正中央并恢复默认显示比例
 - Zoom In & Zoom out：相应调整画布显示比例

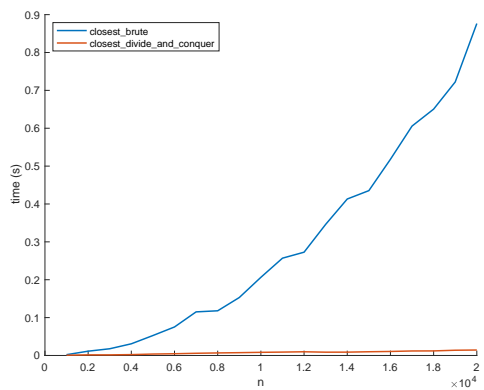


Figure 1: 输入规模 $n \leq 20000$

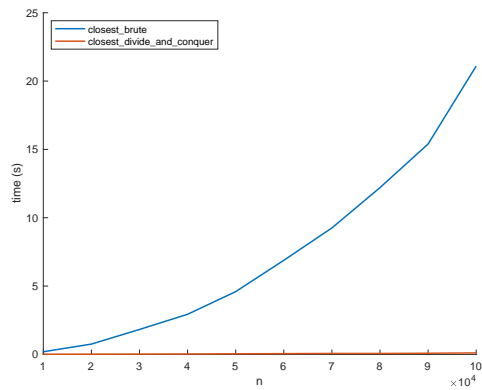


Figure 2: 输入规模 $n \in [10^4, 10^5]$

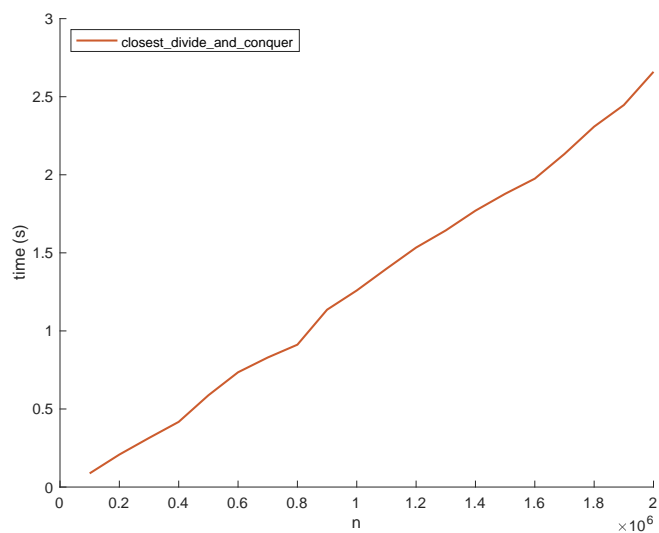


Figure 3: 输入规模 $n \in [10^5, 2 \times 10^6]$