

梯形图(trapezoidal map)实验

一、 实验目的

1. 阅读了解点定位问题的资料，对梯形图的形成初步理解；
2. 通过实验，实现梯形图结构的构造及查询算法，并通过对它的性能分析，了解此算法的优劣。

二、 实验环境

WinXP Visual Studio 2008

三、 实验分析

点定位问题

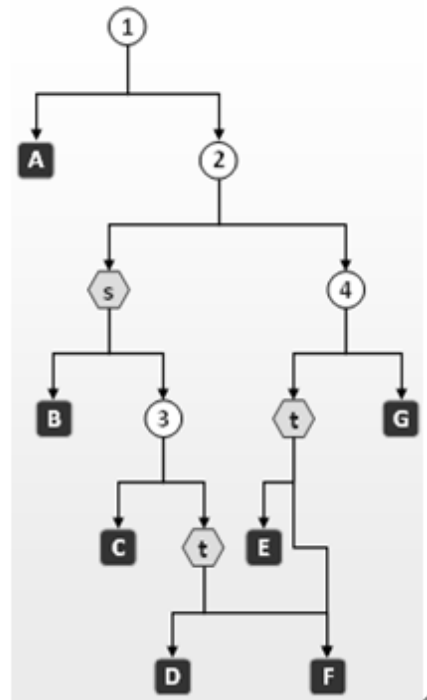
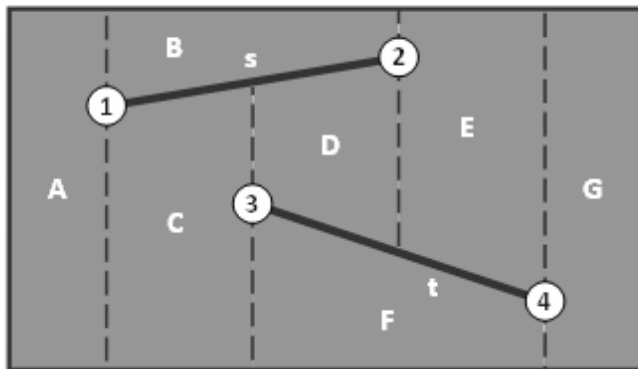
给定一定数目的曲线，通过一定的处理构建一个数据结构，这样对于给定的任何一点可以有效地找到和这个点对应的单元。

实现以上算法一个比较有效的方法是将搜索区域按照空间分成垂直的条带状，并对于每个带内部的进行查找，这样在 X 和 Y 方向上同时进行二分查找，可以在 $O(\log n)$ 的时间内实现点的查询，只是这种算法需要的空间复杂度为 $O(n^2)$ 。

在对以上算法改进的过程中，产生一种随机插入构建 (randomized incremental construction) 思想而提出的算法，算法将搜索结构分解成一些竖直的梯形，被称为梯形图。梯形图结构的构造与查询是通过建立一个随机增量式算法 (randomized incremental algorithm)，利用它为任意一组线段集 S，构造出对应的一个有向循环图的查找结构梯形图 T(S)，在构造梯形图过程中，同时构造一个数据结构 D，借助它在 T(S) 中进行点定位查询。该算法空间复杂度为 $O(n)$ ，查询的时间复杂度的期望值为 $O(\log n)$

该算法所构造的数据结构 D 是有向循环图的查找结构，其中有唯一的根节点，同时对应于 S 的梯形图中的每个梯形，有且仅有一个叶子。每个内部节点的度为 2。所有内部节点分为两类：x-节点和 y-节点。每个 x-节点用 S 中某条线段的一个端点作为判断准则；每个 y-节点用 S 中某条线段作为判断准则。

如下图所示，圆形代表 x-节点，六边形代表 y-节点，黑色正方形是叶节点：



在对点 p 进行查询时，从根节点出发，沿着某条有向路径，每遇到一个新节点都将其与 p 比较，若是 x -节点，取该点横坐标与 p 横坐标比较，若 p 较大则选右子树，否则选左子树，若是 y -节点，取 p 投影到节点所存线段的点与 p 比较， p 在下面选左子树，否则选右子树。最终到达的那个叶子就对应于 $T(S)$ 中包含 p 的那个梯形。

四、 实验算法

我们按照课件上给出的构造查找结构的递增式算法

Algorithm ConstructTrapezoidalMap(S)

1、构造一个包围框 R 包含 S 中所有线段;

2、将 S 中线段顺序打乱

2、 $TM(S)$ = 单个梯形图 R ,

$SS(S)$ = 单个表示 R 的根节点;

3、对每条属于 S 的线段 $s=pq$ ，找到 p 所在的梯形令其为 $Trap$ ，做循环操作

/* (1) 对梯形 $Trap$ 进行替换操作，并更新 $TM(S)$ 和 $SS(S)$;

(2) 比较 $Trap$ 右端点 $rightp$ 与线段 s 的位置，若 $rightp$ 在 s 的上方

$Trap=Trap$ 的右下方梯形，

否则 $Trap=Trap$ 的右上方梯形

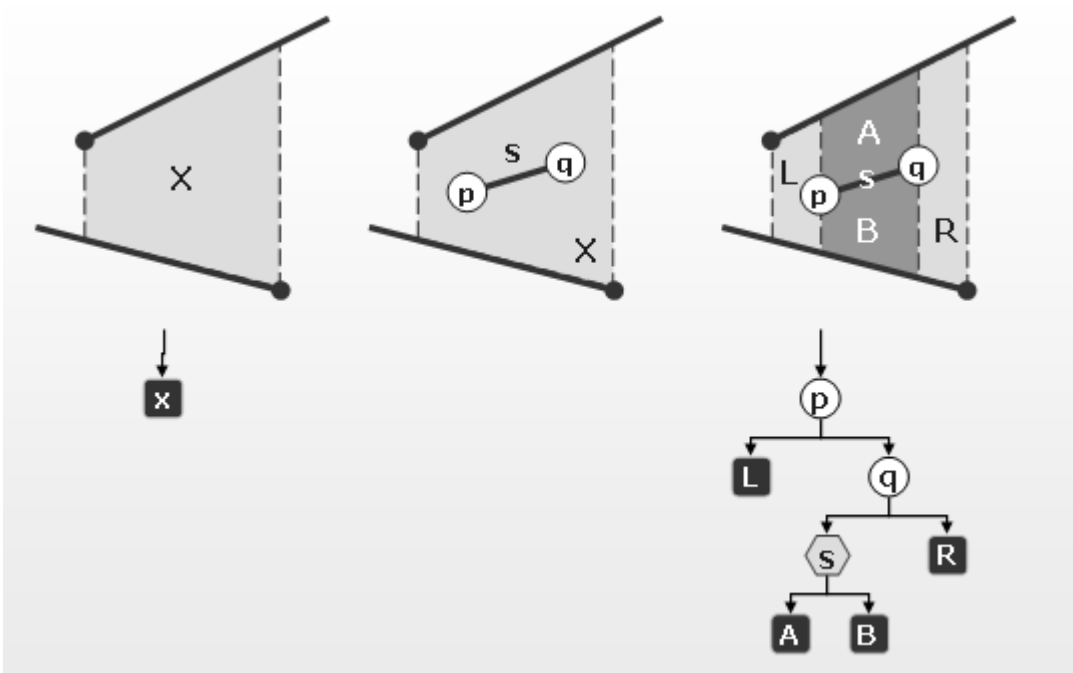
直至 $q < Trap->rightp$ 为空。*/

替换操作

总共有三种情况:

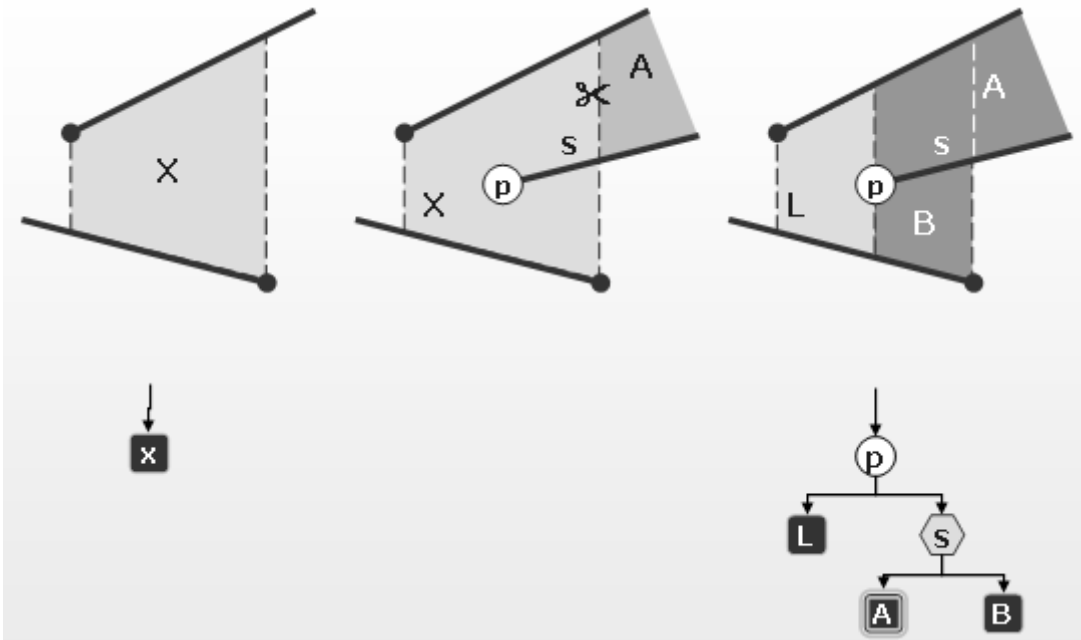
1. 两端点均在 $Trap$ 中

这种情况最容易做，修改 $TM(S)$ 中 $DCEL$ 结构，并替换 $SS(S)$ 中叶节点 X

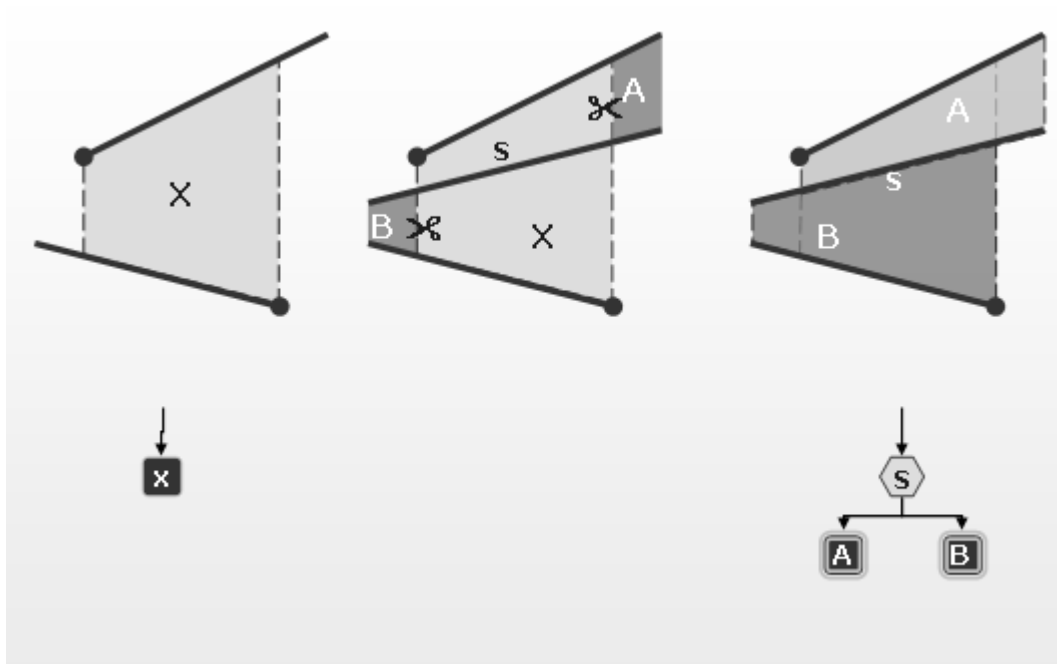


2. 一个端点在 Trap 中

不妨令那个端点是 p , 其中判断哪个叶节点是有两个指向的需要比较 s 与 $\text{Trap} \rightarrow \text{right}$ 位置



3. 无端点在 Trap 中



五、 程序功能

1、 界面操作

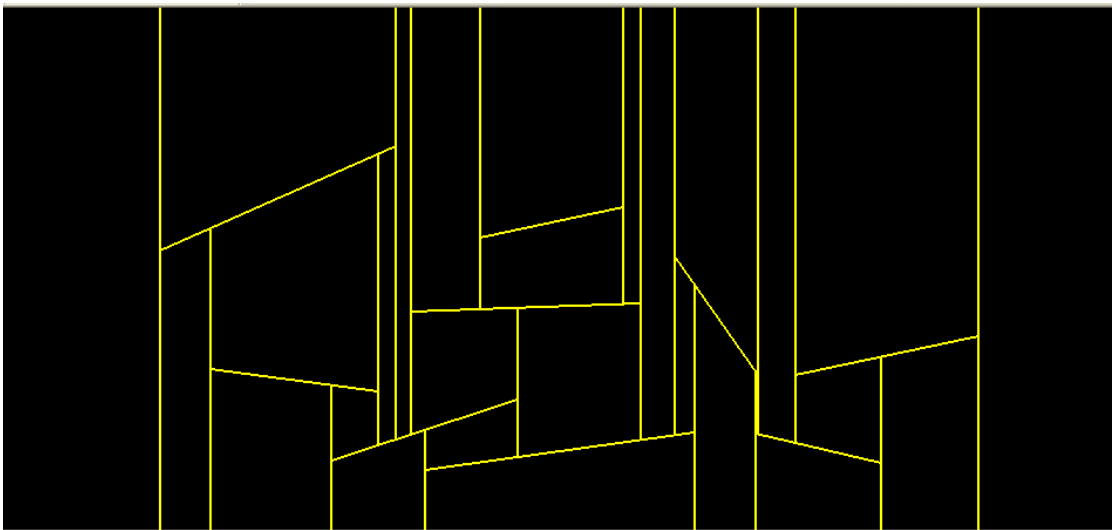
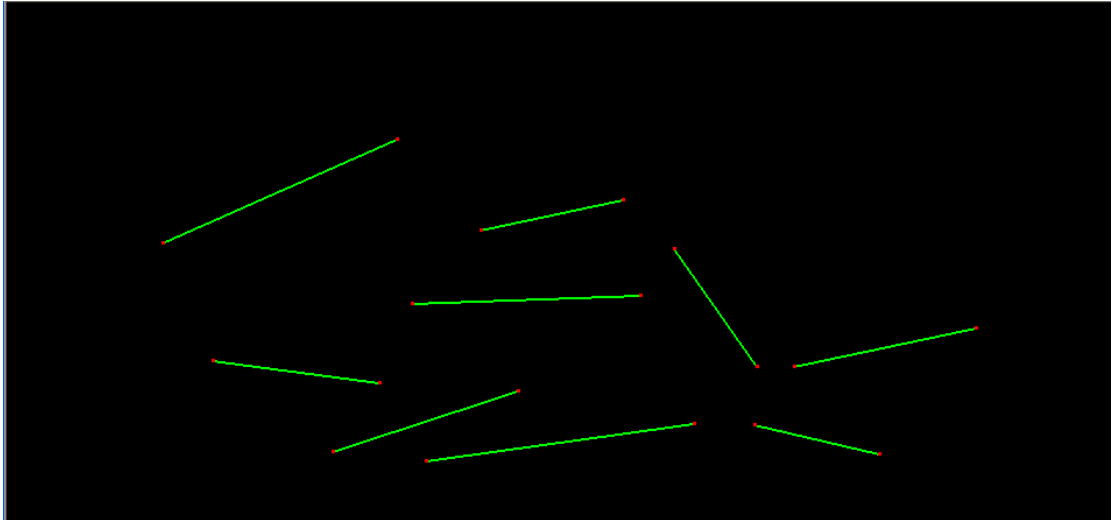
- 鼠标左键单击添加点
- 鼠标左键拖动已有线段
- 鼠标左键拖动点
- 鼠标左键双击删除线段
- 鼠标左键选定区域放大
- 鼠标左键移动屏幕

以上操作通过菜单-界面操作里的各项菜单实现。

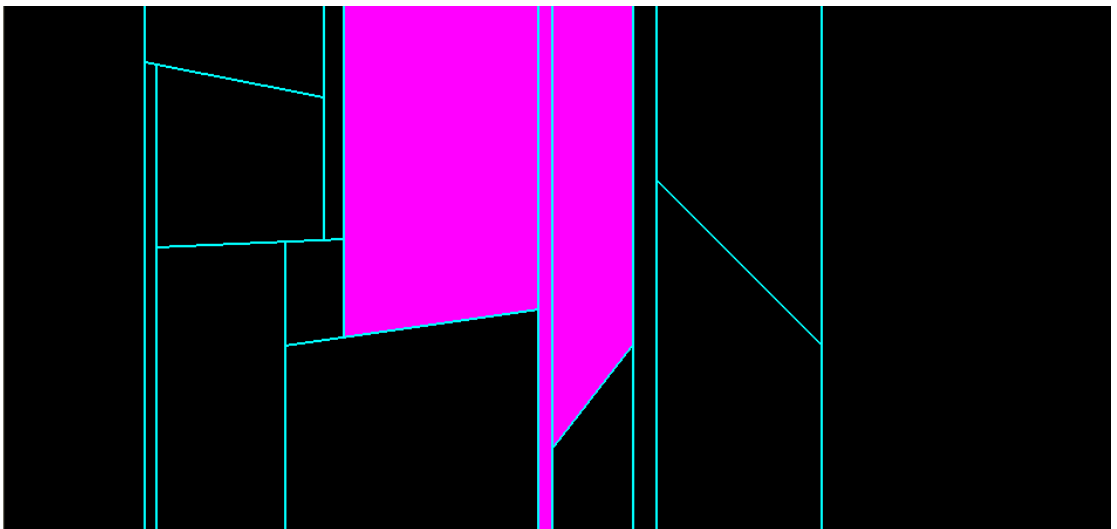
2、 梯形图操作

- 随机生成 n 条线段

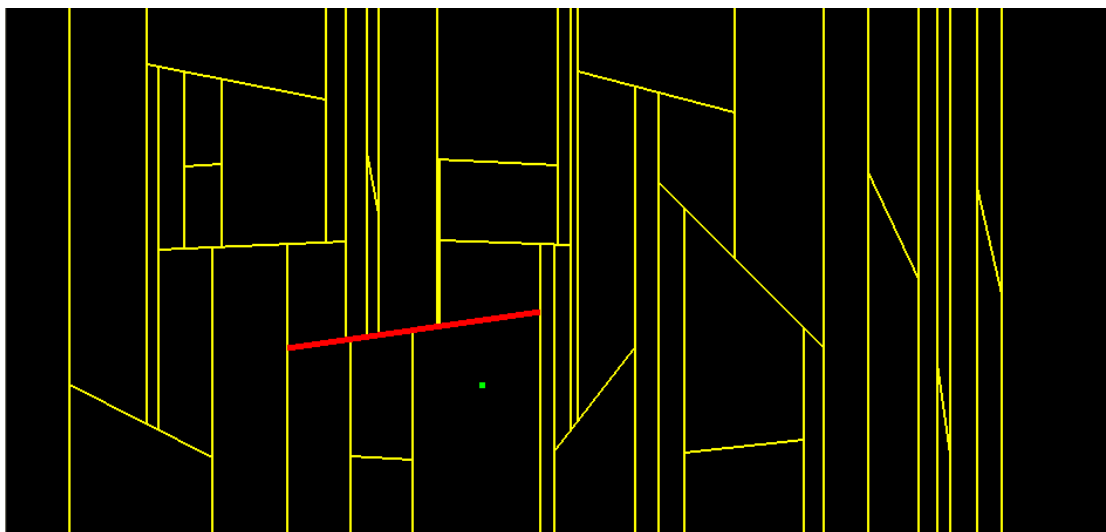
直接生成梯形图



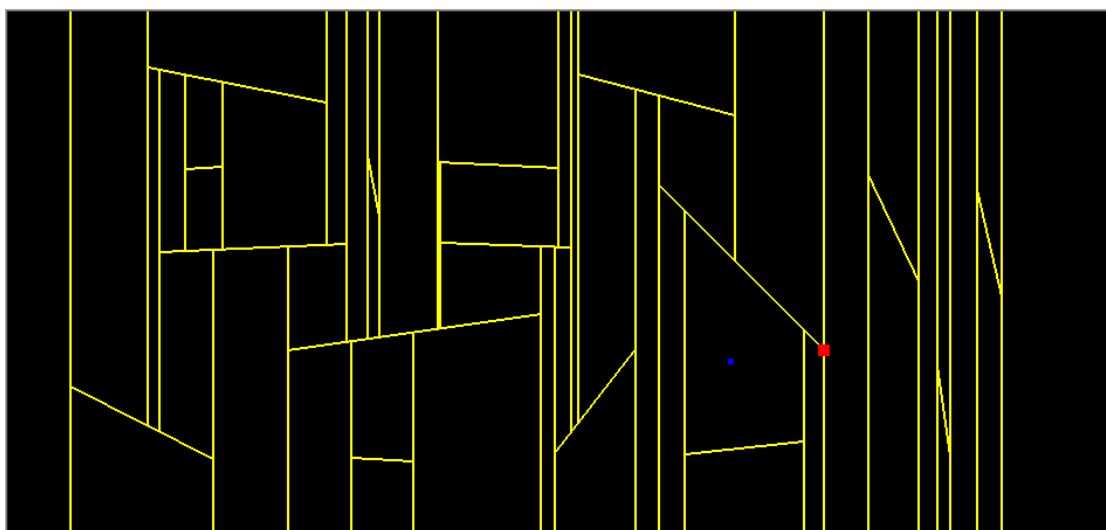
演示梯形图生成过程



直接进行点定位



演示点定位的过程



对上述数据分析可以发现构造梯形图所画时间基本在 $O(n \log n)$ 对点的点定位查找, 基本可以在 $O(\log n)$ 的时间内完成

六、 实验收获

通过本次实验, 我们对梯形图的生成算法有了很好的理解。这是本次课程最后一次实验, 因为临近考试, 所以时间显得很紧张, 有些想法没能实现, 稍微有些遗憾。但总的来说, 我觉得上了这门课以后收获非常大。计算几何这门课与其它算法类课程有一个非常明显的区别: 它需要考虑更多的实际情况。以梯形图为例, 它需要考虑共线垂线段等各类情况, 所以更注重实际一些。

最后感谢两位老师本学期精彩的讲课!