Computational Geometry CSL 852
Lecture 27
Topic: Range Searching

<u>Claim</u>: Reporting range query for orthogonal rectangles takes $O(\sqrt{n} + \underline{m})$ steps using a k-d tree (2-d-tree in two dimensions).

m: output size

Suppose $Q(n)$ is the worst case query time (# nodes visited) for n points

$$Q'(n) = 2 \cdot Q'\left(\frac{n}{4}\right) + 2$$

nodes visited

$$\Rightarrow Q'(n) \text{ is } O(\sqrt{n})$$

$Q'(n)$: # of rectangular regions an <u>edge</u> intersects, where the rectangular region contains n points

$$Q(n) \le 4 \cdot Q'(n)$$

K-d tree is for K dim range search

$Q_k(n) = ? \quad O(n^{1-\frac{1}{k}})$

$S_2(n) = O(n)$ – a point is stored exactly once

$S_k(n) = ? \quad k \cdot n$ (is

Can we improve the $\sqrt{n}$ bound?

Perhaps space must increase

Inherent space-time tradeoffs must be lower bounds

Build a 1-dimensional K-d tree.

Observation: Any arbitrary interval $[x_1, x_2]$ can be written as a union of $\boxed{2\log n}$ disjoint "canonical" intervals

A two dimensional <u>range tree</u> is a nested data structure where the primary tree is built on x coordinates and within each node we have a 1 dimensional range-search data structure on y coordinates.

Query time : $O(\log^2 n + k)$

Space : $O(n \log n)$