CG
10/16/08

Computing a Trapezoidal Decomp

In General: Input is n line segments S

To start: 1) S forms a PSLG of bounded degree
2) no 2 endpoints have same X value.

Random Incremental

Alg Trap Map (S)

Input: Segments $S = s_1 \cdots s_n$

Output: Trap Map τ & Point location DS $\mathcal{D}$.

1) Init: Build bounding box B & $\mathcal{D}$ for B

2) Randomly order $s_1, \cdots, s_n$

3) for $i = 1$ to n.

a) Find trap containing left-endpoint of s_i

b) Stitch in s_i from left-to-right.

c) Update $\mathcal{D}$.

Following pointers

1) left-vertical (Δ)
2) right-vertical (Δ)
3) left-trap (edge)
4) right-trap (edge)

Thm Trap Map

1) $O(n \log n)$ expected time
2) $|\mathcal{D}| = O(n)$ expected
3) search time is $O(\log n)$ expected

Expectation is over $n!$ orderings of S

3) Query Time

a) Each update to $\mathcal{D}$ increase depth by at most 3.

b) Fix query point $x \in B$

Goal: consider random variable

$X = \text{depth}(x, \mathcal{D})$. Bound $E(X)$

Consider X_i = # levels added when adding S_i

$$X = \sum X_i$$

$$E(X) = \sum E(X_i)$$

Use Backwards analysis

---

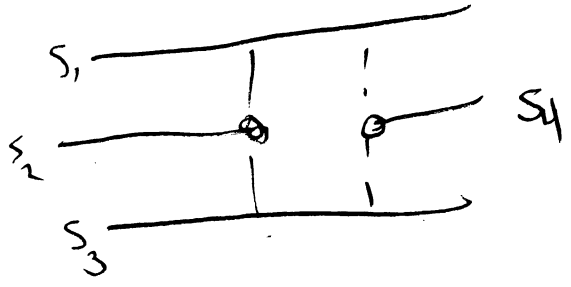
note $E(X_i) \leq 3P_i$

P_i = prob that trap Δ_i is removed

$x \in \Delta_i$

Claim $P_i \leq 4/i$

Δ_i is removed if one of 4 segs changes.



$E(X_i) \leq 12/i$ $\quad E(X) \leq 12 H_n$

H_n harmonic number.

2) The size of $\mathcal{D}$

node = # Traps + internal nodes.

Traps = $O(n)$ (Thm)

K_i = # new traps at time i.

K_{i-1} = # internal nodes

Back ward analysis

K_i = # traps removed when removing random seg

traps = $3i+1$

$p_i \leq 4/i$

$E(K_i) \leq (3i+1)\left(\frac{4}{i}\right) = O(1)$

$E(|\mathcal{D}|) = \sum E(K_i) = O(n)$