Lect. #31: Range Searching

(S, R)

Orthogonal range searching

R: set of rectangles

set of triangles, disks, halfplanes, ....

\((X, \oplus)\) : semigroup

\(w : S \rightarrow X\)

Preprocess \(S\) into a data structure s.t.

for a query region \(R \in \mathcal{R}\), compute

\[ \bigoplus_{p : p \in \mathcal{R}} w(p) \]

Range counting

\(X = \mathbb{N} \oplus = +\)

\(w(p) = 1 \quad \forall p \in S\)

Emptiness query

\(\text{yes if } \exists \mathcal{R} \ni S \neq \emptyset\)

\(X = \{0, 1\} \quad \oplus = V\)
Range Reporting

\[ X = 2 \quad \Theta = U \]

\[ W(p) = \{ p \} \]

Orthogonal

\[ Q(n) : \text{query time} \quad O(5n) \quad O(\log n) \]

\[ S(n) : \text{size} \quad O(n) \quad O(n \log n) \]

\[ P(n) : \text{preprocessing time} \]

Semigroup model

\[ f = \{ C_i, \ldots, C_m \} : \text{canonical subsets} \]

\[ \cdot C_i \subseteq S \]

\[ \cdot \forall v \in \mathbb{R} \]

\[ f \supseteq f(v) = \{ C_i, \ldots, C_m \} \]

\[ * C_i \cap C_j = \emptyset \quad \forall C_i, C_j \in f(v) \]

\[ * \cup C_i = S \cap \mathbb{R} \]

\[ a \in f(v) \]

Minimize

\[ \text{Size} \geq m \quad Q(n) \geq \max_{v \in \mathbb{R}} |f(v)| \]
Filtering search

Range reporting

$R_M = O(k)$

Query procedure can use $O(k)$ time to guide the search.

Halfplane Range Reporting

Given a query line $l$, report $SN(l)$, all points that lie above $l$. 
points in $S$

Suppose $S$ are in convex position

Intersect ($P, l$)

Find $p, q = l \cap P$

March from $p$ to $q$ along $P$

(2) report all points in $l \cap AS$.

Time = $O(\log n + k)$

- Compute convex layers of $P_0, P_1, \ldots, P_m$ of $S$

  $S_\geq S_i$: set of points on $P_i$

  $\text{size} = O(n) \quad P(n) = O(n \log n)$

- Query procedure

  $i = 0$

  while $p, q = \text{Intersect} (l, P_i)$

  Report ($p, q, P_i$)

  $i = i + 1$

  end while
Query time:
Suppose \( L \) intersects \( \mathcal{P}_0 \ldots \mathcal{P}_y \) and does not intersect \( \mathcal{P}_{R+1} \)

\[
(y+1) \cdot \log n + R \leq R
\]

\[
Q(y) = O((y+1) \cdot \log n)
\]

Fractional cascading:

\[
O(\log n + R)
\]

Halfplane Range Counting:

\[
Q(y) = O(\log n) \quad S(y) = O(n^2)
\]

\[
Q(n) = O(n \sqrt{n \log n}) \quad S(n) = O(n)
\]
Primal plane

\[ p = (a, b) \rightarrow p^*: y = ax + b \]

Dual plane

\[ l: y = \alpha x + \beta \rightarrow l^* = (-\alpha, \beta) \]

Primal

Dual

Report all lines lying above \( l^* \)
Size: $O(n^2)$
- Compute $A(s^*)$  $\text{Time: } O(n^2)$

$O(n^2)$:
- For each face $f \in A(s^*)$
  - $W_f$: # lines lying above $f$ $O(n^2)$

$O(n^2)$: Preprocess $A(s^*)$ for point-location queries $O(n^2 \log n)$

Query procedure
- Given $l$, locate the face $f(l)$ containing $l^*$

$O(\log n)$
- Report $W_{f(l)}$