CS 594: Representations in Algorithm Design **Spring 2022**

Lecture on 04/07/2022

Lecturer: Xiaorui Sun *Scribe: Shahrukh Haider*

1 Geometric Data

Geometric data is a set of points where each points can be represented by a vector of dimension d

P = { x : x R^d }

Example: Google maps can be represented by geometric data for d = 2

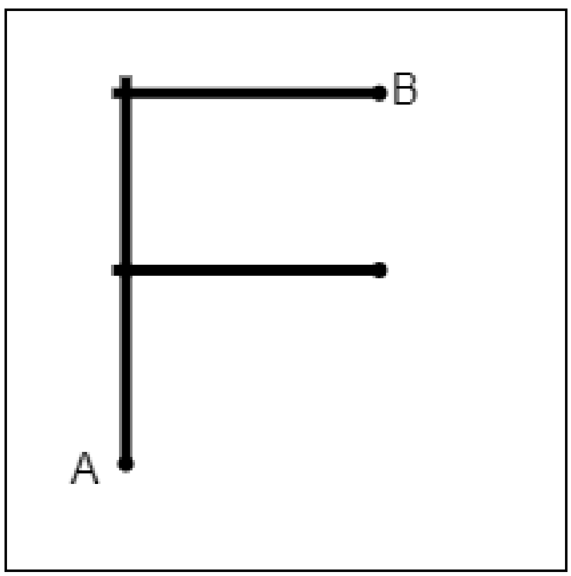

Figure 1: Google map representation

Problem: We have to calculate the distance between two points A and B. We could use Djikstra's algorithm for finding the shortest path between A and B.
Here, we don't have to search in all direction as data is geometric in nature. We can

do a guided search by moving in the direction of second point.

Geometric data can be represented as a graph.
$G_p = (p, E, w)$
Where,
p = points in the graph
E = $p * p$ = number of edges in the graph
w = weight of edges

Here, weight of two points is the distance between two points w(x, y) = d(x, y). Distance between two points can be measured as:
$\| x - y \|_2 = \sqrt{\Sigma(x_i - y_i)^2}\ \forall$ i
or
$\| x - y \|_2 = |\ x_i - y_i\ |\ \forall$ i

Geometric data can be used to solve these problems:

1. Shortest path
2. Clustering problem
3. Routing problem

2 WSPD

2.1 Motivation for WSPD

We have n points and graph construction takes $\Theta(n^2)$
Goal: We have to reduce the time complexity while preserving pair-wise distance between two points

2.2 $1/\epsilon$ - Well Separated Pairs Decomposition

Here, we group the points such that points in a group are much closer together and are far from points in other groups.
So if we know the distance between two points in different groups then we can approximate the distance between any two points between these two groups. d($x \to G_1$, $y \to G_3$) $\approx$ d($x^{'} \to G_1$, $y^{'} \to G_3$)

Definition : WSPD is a set of pairs $\{(A_i,\ B_i)\}$ satisfying

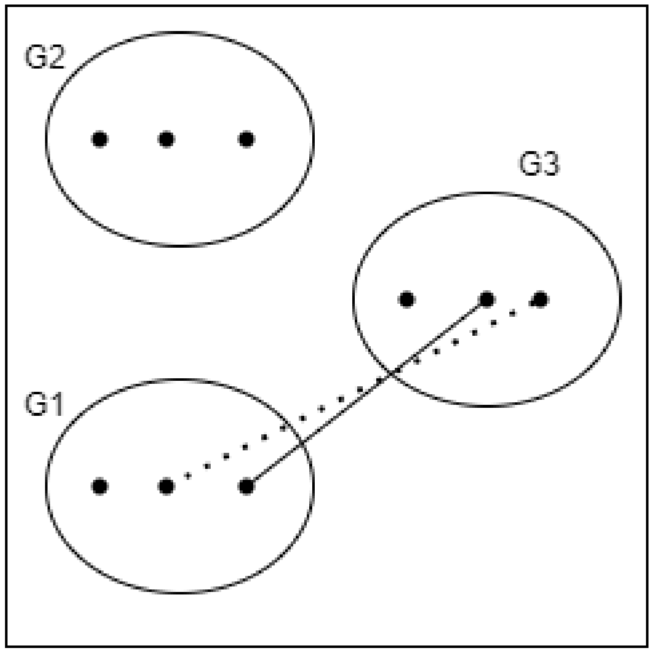

Figure 2: WSPD distance approximation

1. $A_i, B_i \subseteq$ P $\forall$ i

2. $A_i \cap B_i = \phi$

3. $A_i * B_i = P * P \setminus (x, x)_i \in P$

   Here, A * B is defined as:
   $A * B = (x, y) : x \in A, y \in B$

4. A_i and B_i are $1/\epsilon$ separated $\implies$ max{diam(A_i), diam(B_i)} $\leq \epsilon$. d(A, B)
   Here, diam(A_i) = max d(p, q), where p, q $\in A_i$
   and d(A, B) = min d(p, q) where p $\in A_i$ and q $\in B_i$

2.3 Application of WSPD

Closest Pair Input: p $\subseteq R^d$
Output: (x, y) $\in$ P such that d(x, y) is minimised
d(x, y) $\leq$ min d(x', y') where x', y' $\in$ P and $x' \neq y'$

for 2 dimensions this problem can be solved in O(nlogn) time complexity without WSPD

WSPD algorithm:

    d $\leftarrow \infty$
    for pair(A_i, B_i) $\in$ WSPD
        if | A_i | > 1 or | B_i | > 1 pass
        else d $\Leftarrow$ min(d, d(p,q)) where p *in* A_i and q *in* B_i
    return d

Time Complexity:
$1/\epsilon$ - WSPD for d-dimensional space = O(n . log(Φ(P))) ($1/\epsilon^d$)
where,
Φ(P) = max d(p, q) / min d(p, q) such that p, q $\in$ P

2.4 Construction of WSPD using Quad Tree

There are n points on a plane. We consider the plane a square and divide the square recursively into 4 smaller squares until each square contains either zero or one point

Here, depth of the tree $\leq \log(\Phi(P) / \sqrt{d})$
Time Complexity: O(n . log(Φ(P))) / $\sqrt{d}$) 2^d)

Algorithm to construct WSPD with quad tree

WSPD(S_1, S_2) {
    if diam(S_1) < diam(S_2)
        swap(S_1, S_2)
    if diam(S_1) $\leq$ d(S_1, S_2) . ϵ
        return (S_1, S_2)
    otherwise
        WSPD(S_1', S_2) where S_1' is a sub-sequence of S_1
}

In the next lecture, we will conclude our discussion on WSPD.

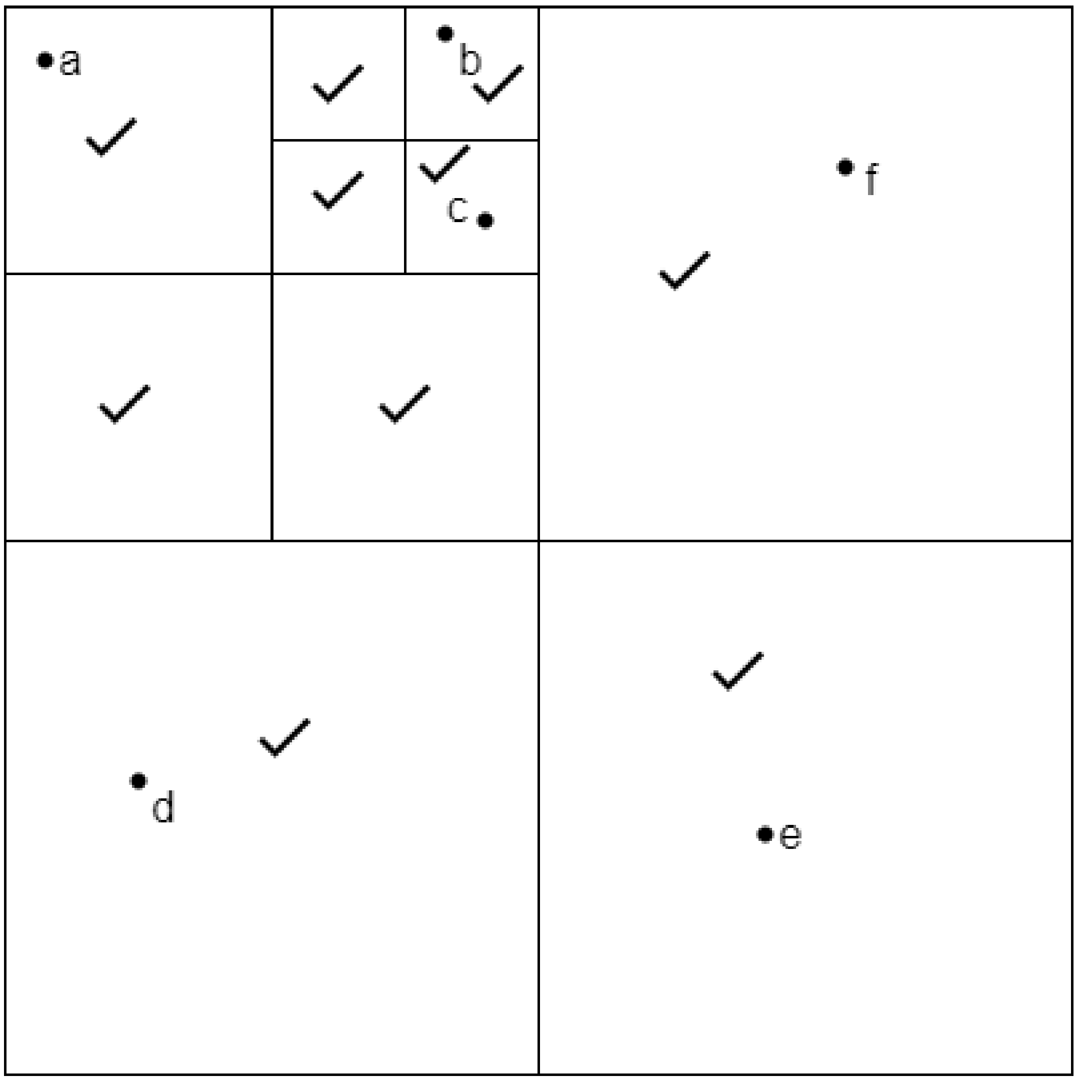

Figure 3: Quad Tree representation

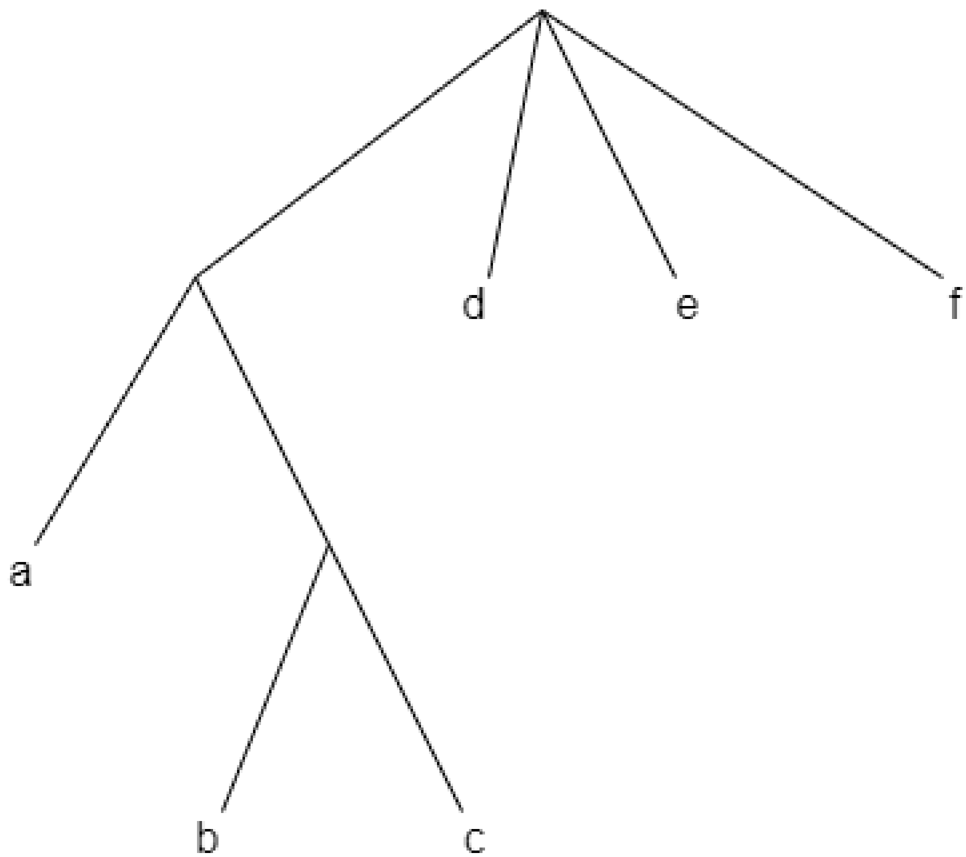

Figure 4: Quad Tree