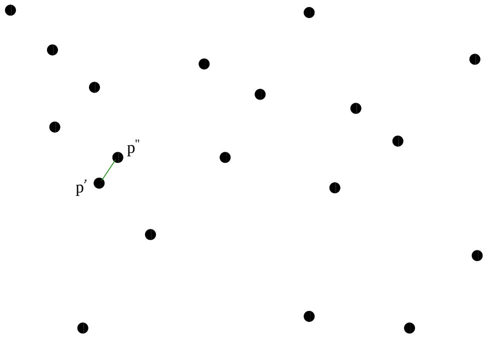
Paradigm: Divide-and-Conquer

Finding a Closest Pair of Points

R. Inkulu
http://www.iitg.ac.in/rinkulu/

Definition

Find a closest pair of points in the given set S of n points in R^2.

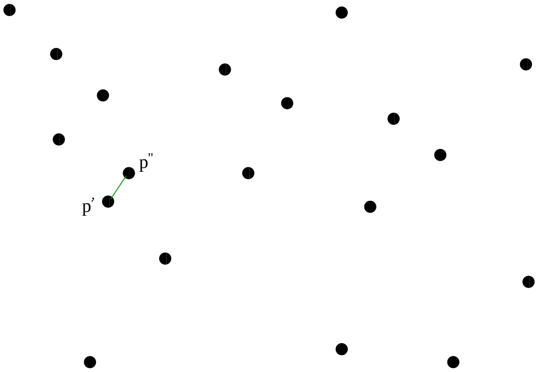


closest pair of points are p' and p''

for convenience, assume that no two points in S are having the same x- or y-coordi

Brute-Force Algorithm

Computing the distance between every two points in S, and choosing a pair with the smallest distance: $O(n^2)$ time.



Divide-Conquer-Combine

recursively do the following:

(1) partition points into left and right halves

(2) conquer the sub-problems in both the halves

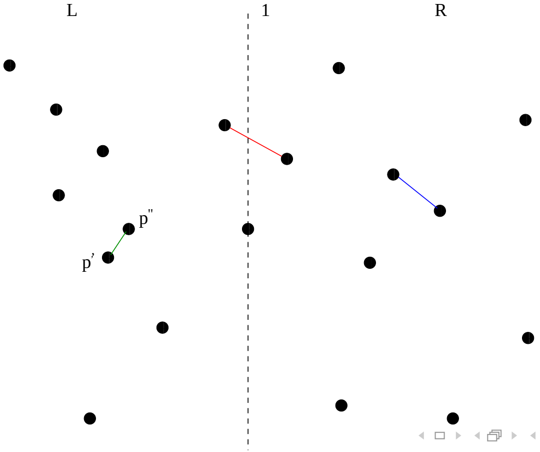
(3) combine the solutions for halves $\leftarrow$ critical

Combining solutions

(i) closest pair is from left half of points

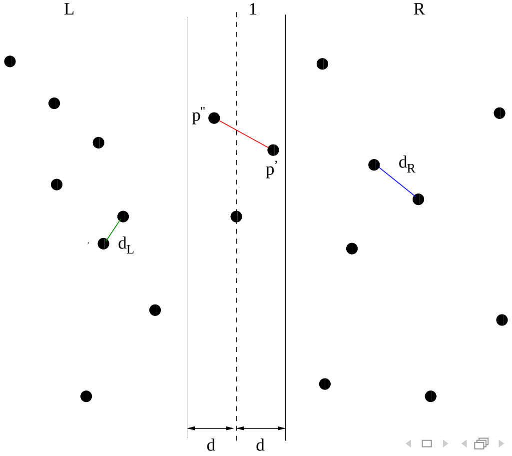
(ii) closest pair is from right half of points

(iii) closest pair is formed by one point from left half and another from right half *leftarrow* how to do in $O(n)$?



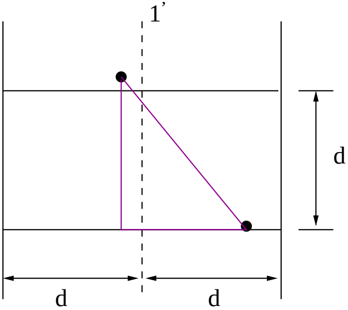
Vertical strip containing p' and p''

Let d_L be the shortest distance between all the points of set L, and let d_R be the shortest distance between all the points of set R. Also, let $d = min(d_L, d_R)$. Then for the Case 3 to occur p' and p'' must lie in the vertical strip of width $2d$ that has the dividing line between L and R at its center.



Horizontal strip containing p' and p''

For every point p' falling in vertical strip, for any point p'' belonging to the vertical strip to be at a distance less than d from p', point p'' must lie within the horizontal strip of height d with base of that strip passing through p'.

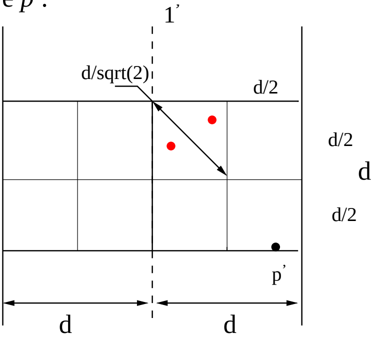


Euclidean distance between p' and p'' is *not* less than d

Bounding the number of candidate p''s w.r.t. a p'

Let us denote the intersection of the horizontal and vertical strips as a rectangle B. Also, G be the regular grid associated with B such that each cell of G is of size $d/2 \times d/2$.

- From last slide, if p'' exists, it belongs to the set of points in B.
- There can be at most one point in each cell of G.
- In the y-sorted list of points falling in vertical strip, p'' is at most seven positions above p'.



Euclidean distance between red points is less than d contradicts

$$d = min(d_L, d_R)$$

Combine step detailed

(a) let S' be the y-sorted list of points falling within the vertical band
for each point p, find the distance between p and each of the next 7 points in S'

(b) let p' and p'' be the least among the search
return $min(dist(p', p''), d_L, d_R)$ with the corresponding points

Correctness

- termination is ensured as each branch of the recursion terminates with at most 3 points
- algorithm is correct: use induction

Analysis: time complexity

initial sorting according to x-coordi: $O(n \lg n)$

initial sorting according to y-coordi: $O(n \lg n)$

$T(n) = 2T(n/2) + O(n)$: $T(n) = O(n \lg n)$

- forming x-sorted and y-sorted lists for recursive invocations: $O(n)$
- identifying points that fall within the vertical strip: $O(n)$
- forming y-sorted list of points that fall within the vertical strip: $O(n)$
- for each p' within the vertical strip, searching for p'' consumes *seven* primitive operations. When all p' kind of points considered together: $7n$

total: $O(n \lg n)$ time