Chapter 6: Prune and Search (Supplementary)

Chin Lung Lu

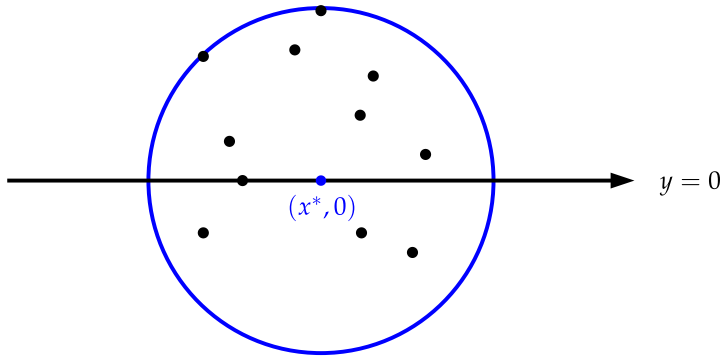
Department of Computer Science

National Tsing Hua University

General 1-center problem

Center of optimum circle lying on $y = 0$

- Imagine that we have a set of points and a line $y = 0$ as follows.

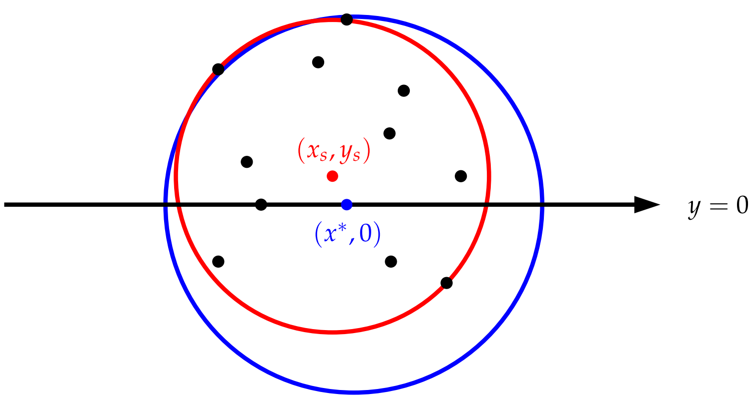


- By using the constrained 1-center algorithm, we can determine the exact location of x^* on this line.

General 1-center problem (cont'd)

Center of optimum circle lying on $y = 0$

- Let (x_s, y_s) be the center of the optimum circle containing all points.



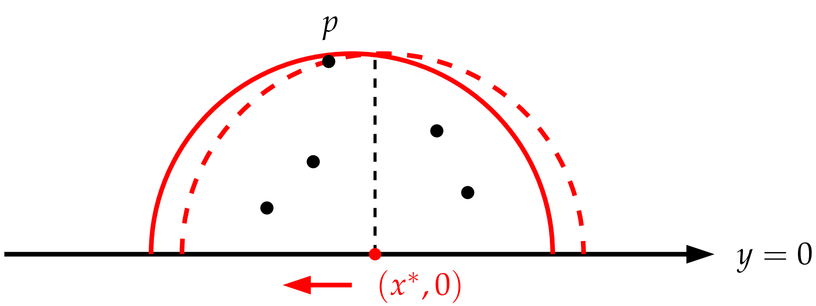
General 1-center problem (cont'd)

Center of optimum circle lying on $y = 0$

- Actually, using the information of x^*, we can determine whether $y_s > 0$, $y_s < 0$ or $y_s = 0$.
- By the same reason (i.e., using the information of y^* obtained by solving the constrained 1-center problem on the line $x = 0$), we can also determine whether $x_s > 0$, $x_s < 0$ or $x_s = 0$.
- Let I be the set of points which are farthest from $(x^*, 0)$.

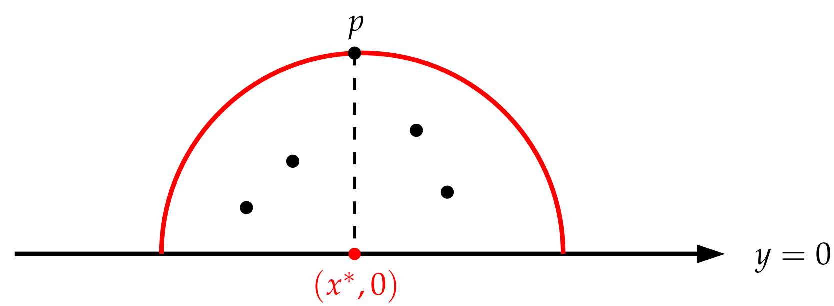
Case 1: I contains only one point p

- In this case, the x-value of p must be equal to x^*.
- Otherwise, we can move x^* towards p along the line $y = 0$.
- It contradicts with our assumption that $(x^*, 0)$ is an optimal solution on the line $y = 0$.



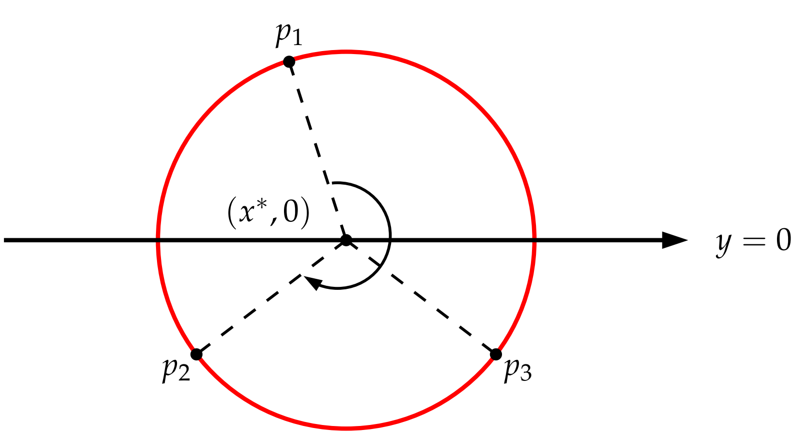
Case 1: I contains only one point p (cont'd)

- In other words, if p is the only farthest point of $(x^*, 0)$, then its x-value must be equal to x^*.
- Thus, we can conclude that y_s has the same sign as that of the y-value of p.



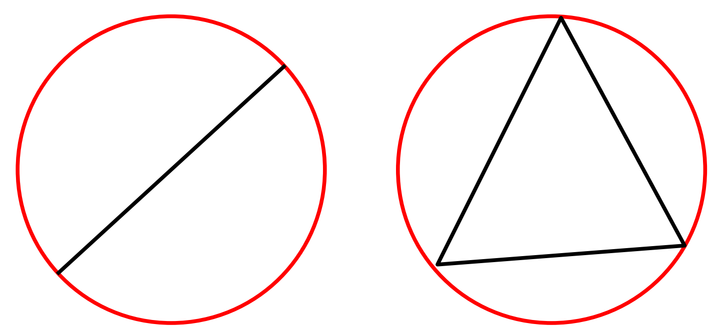
Case 2: I contains more than one point

- Find the smallest arc spanning all the points in I.
- Let the two end points of this arc be p_1 and p_2.
- If this arc is of degree greater than or equal to $180°$, $y_s = 0$.



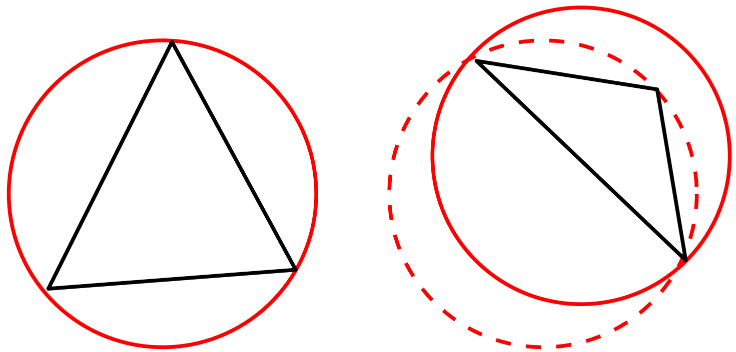
Case 2: I contains more than one point (cont'd)

- Note that a smallest circle containing a set of points is defined by either two points or three points of this set.



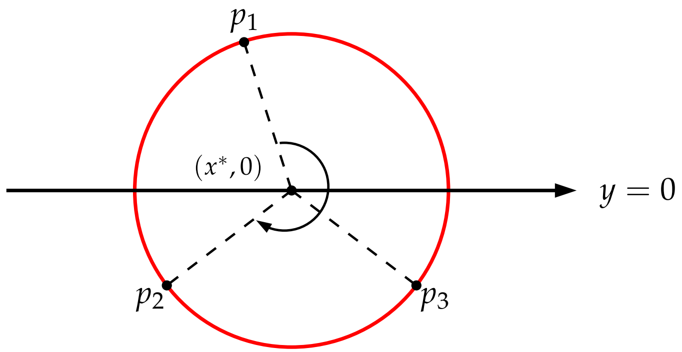
Case 2: I contains more than one point (cont'd)

- Three points define the boundary of a smallest circle enclosing all these three points if and only if they do not form an obtuse triangle (left figure).
- Otherwise, we can replace this circle by using the circle with the longest edge of this triangle as the diameter (right figure).



Case 2: I contains more than one point (cont'd)

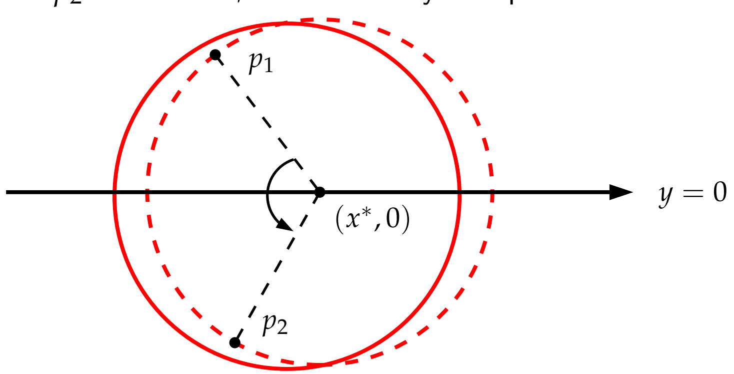
- If the degree of the arc spanning all farthest points is $> 180^\circ$, then there must be at least three such farthest points and these three points do not form an obtuse triangle.



- It means that the present smallest circle is already optimal and hence we conclude that $y_s = 0$ (which is also true when the degree of the arc spanning all farthest points is 180°).

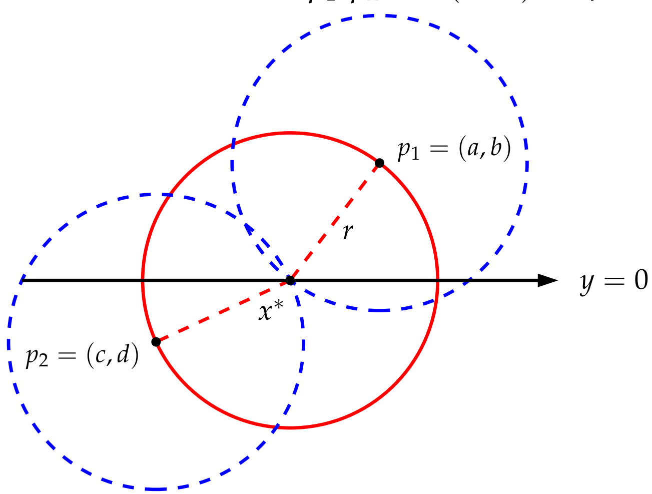
Case 2: I contains more than one point (cont'd)

- Suppose that the arc spanning all farthest points is of degree less than 180°.
- The x-values of end points p_1 and p_2 must be of opposite signs of x^*.
- Assume otherwise, we can move x^* towards the direction where p_1 and p_2 are located, which actually is impossible.



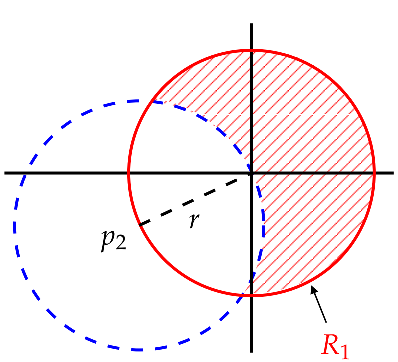
Case 2: I contains more than one point (cont'd)

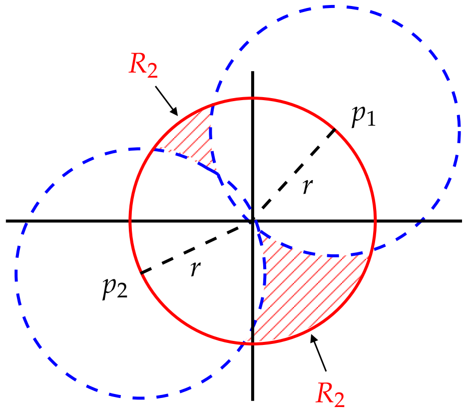
- Therefore, we assume that $p_1 = (a, b)$ and $p_2 = (c, d)$ with $a > x^*, b > 0$ and $c < x^*, d < 0$, and use r as the radius to draw three circles centered at p_1, p_2 and $(x^*, 0)$, respectively.



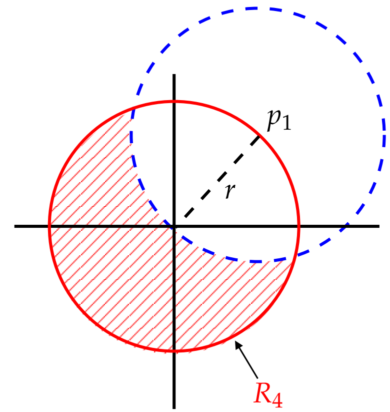
Case 2: I contains more than one point (cont'd)

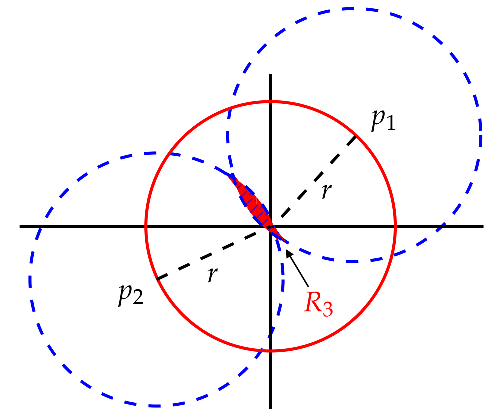
- Through these three circles, we can find four regions in the circle centered at $(x^*, 0)$: R_1, R_2, R_3 and R_4.





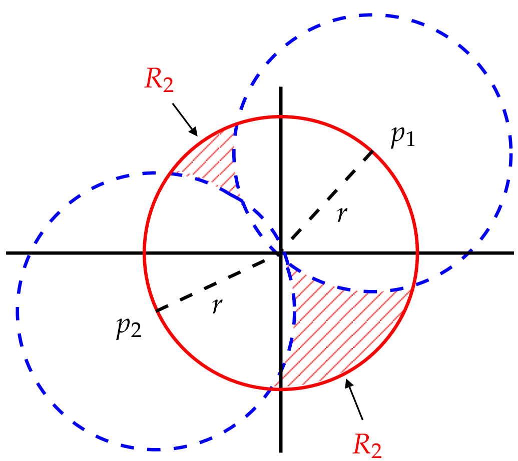
Case 2: I contains more than one point (cont'd)





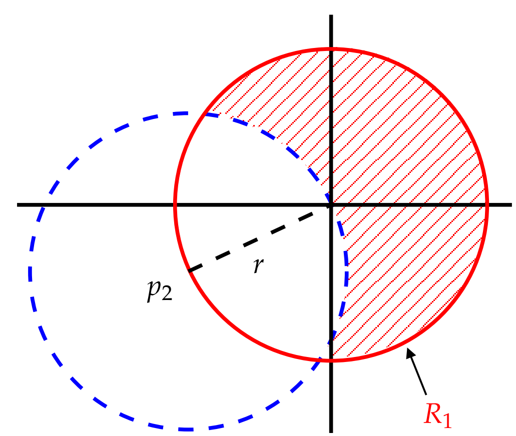
Case 2: I contains more than one point (cont'd)

- Clearly, the distance between any point x in R_2 and p_1 (or p_2) is greater than r and hence the center of the optimum circle cannot be located in R_2.



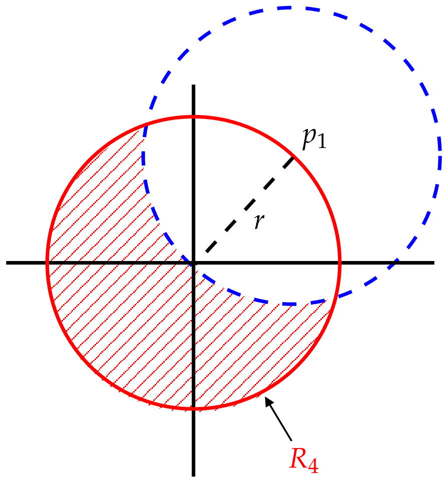
Case 2: I contains more than one point (cont'd)

- The distance between any point x in R_1 and p_2 is larger than r, and hence the center of the optimum circle cannot be located in R_1.



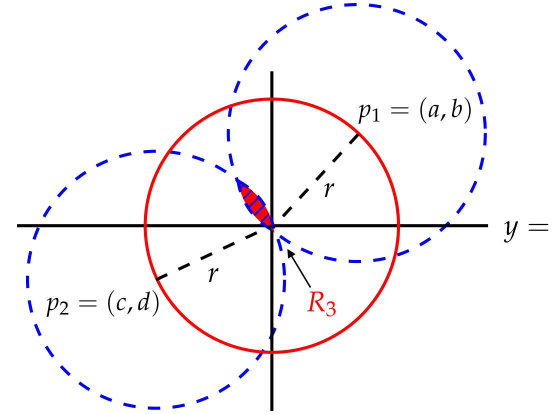
Case 2: I contains more than one point (cont'd)

- The distance between any point x in R_4 and p_1 is larger than r, and hence the center of the optimum circle cannot be located in R_4.



Case 2: I contains more than one point (cont'd)

- Therefore, the optimum center must be located in region R_3.



- Clearly, moving $(x^*, 0)$ towards the midpoint of the line $\overline{p_1 p_2}$ (which is in R_3) will obtain a smaller circle to cover all points.
- It implies that the sign of y_s must be the sign of $\frac{b+d}{2} = \frac{y_1+y_2}{2}$.

Procedure 6-2

Procedure 6-2:
Input: A set S of points, a line $y = y^*$ and (x^*, y^*), where (x^*, y^*) is the solution of the constrained 1-center problem for S.
Output: Whether $y_s > y^*, y_s < y^*$ or $y_s = y^*$, where (x_s, y_s) is the optimal solution of the 1-center problem for S.

1. Find I which is the set of points that are farthest from (x^*, y^*).
2. Case 1 I contains only one point $p = (x_p, y_p)$.
   If $y_p > y^*$, report $y_s > y^*$ and exit.
   If $y_p < y^*$, report $y_s < y^*$ and exit.

Procedure 6-2 (cont'd)

3. Case 2 I contains more than one point.
   In I, find $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$ which are the two end points forming the smallest arc spanning all of the points in I.
   if the degree of this arc is greater than or equal to 180° **then**
     Report $y_s = y^*$ and exit.
   endif
   if the degree of this arc is small than 180° **then**
     Let $y_c = \frac{y_1+y_2}{2}$.
     If $y_c > y^*$, report $y_s > y^*$ and exit.
     If $y_c < y^*$, report $y_s < y^*$ and exit.
   endif

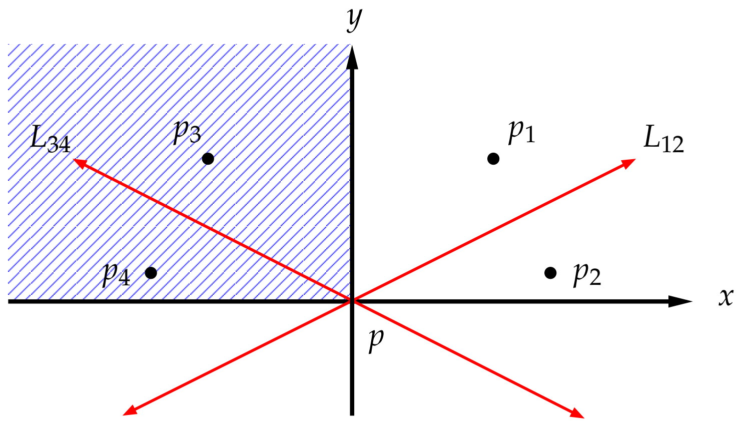
Basic idea of prune and search algorithm

1-center problem

- Let us consider four planar points shown on the next slide.
- For two pairs (p_1, p_2) and (p_3, p_4), we draw bisectors L_{12} and L_{34} of line segments $\overline{p_1 p_2}$ and $\overline{p_3 p_4}$, respectively.
- Let L_{12} and L_{34} intersect at a point, say p.
- Move the origin of the coordinate system to p.
- Also rotate the x-axis such that L_{34} has a negative slope and L_{12} has a positive slope.

Basic idea of prune and search algorithm (cont'd)

1-center problem



- Suppose that the center of the optimum circle must be in the shaded area.

Basic idea of prune and search algorithm (cont'd)

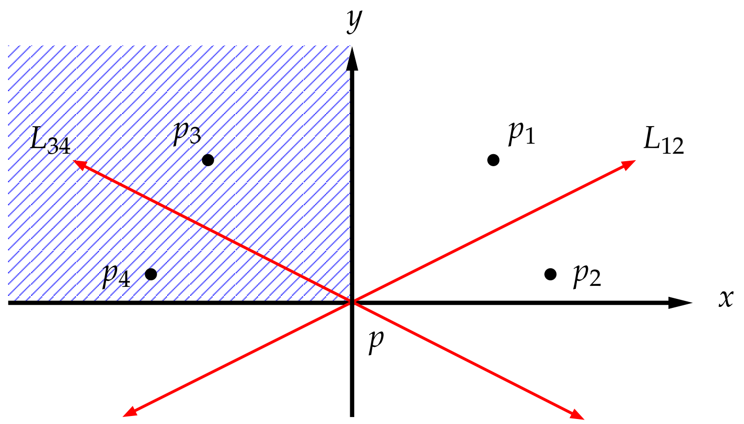
1-center problem

- We apply the constrained 1-center algorithm, requiring that the center be located on $y = 0$.
- After that, we utilize Procedure 6-2 to find out that we should move upward (since we assume $(x_s, y_s) \in$ 2nd quadrant).
- We then repeat this process by applying the constrained 1-center algorithm again to $x = 0$.
- We wll find out that we should move to the left (because we assume $(x_s, y_s) \in$ 2nd quadrant).
- Therefore, the optimal location (x_s, y_s) must be located in the shaded region.
- Because there is one bisection which does not intersect with the shaded region, we can eliminate one point from consideration.

Basic idea of prune and search algorithm (cont'd)

1-center problem

- In our case, L_{12} does not intersect with the shaded region.
- Since p_1 is at the same side of the shaded region, p_1 is nearer to the optimal center than p_2 and hence we can eliminate p_1.



Prune and search algorithm

1-center problem

Algorithm: Prune and search method for 1-center problem
Input: A set $S = \{p_1, p_2, \ldots, p_n\}$ of n points.
Output: The smallest enclosing circle for S.

1. If S contains no more than 16 points, solve the problem by a brute and force method.
2. Form disjoint pairs of points $(p_1, p_2), (p_3, p_4), \ldots, (p_{n-1}, p_n)$.
   For each (p_i, p_{i+1}), find the perpendicular bisector of $\overline{p_i p_{i+1}}$.
   Denote them as $L_{i/2}$ for $i = 2, 4, \ldots, n$.
   Compute the slope s_k of L_k for $k = 1, 2, \ldots, n/2$.
3. Compute the median of s_k's and denote it as s_m.

Prune and search algorithm (cont'd)

1-center problem

4. Rotate the coordinate system so that the x-axis coincides with $y = s_m x$.
   Let the set of L_k's with positive (negative) slopes be I^+ (I^-).
   `/* Note that both of them are of size n/4 */`
5. Construct disjoint pairs of the lines (L_{i+}, L_{i-}) for $i = 1, \ldots, n/4$, where $L_{i+} \in I^+$ and $L_{i-} \in I^-$.
   Find the intersection of each pair of them and denote it by (a_i, b_i).
6. Find the median of b_i's and denote it as y^*.
   Apply the constrained 1-center algorithm to S, requiring that the center of circle be located on $y = y^*$.
   Let this constrained 1-center be (x', y^*).

Prune and search algorithm (cont'd)

1-center problem

7. Apply Procedure 6-2 using S and (x', y^*) as the parameters.
   If $y_s = y^*$, report "(x', y^*) is the optimal solution" and exit.
   Otherwise, report $y_s > y^*$ or $y_s < y^*$.
8. Find the median of a_i's and denote it by x^*.
   Apply the constrained 1-center algorithm to S, requiring that the center of circle be located on $x = x^*$.
   Let this constrained 1-center be (x^*, y').
9. Apply Procedure 6-2 using S and (x^*, y') as the parameters.
   If $x_s = x^*$, report "(x^*, y') is the optimal solution" and exit.
   Otherwise, report $x_s > x^*$ or $x_s < x^*$.

Prune and search algorithm (cont'd)

1-center problem

10. Case 1 $x_s > x^*$ and $y_s > y^*$ `/* 1st quadrant */`
    Find all (a_i, b_i)'s such that $a_i < x^*$ and $b_i < y^*$.
    Let (a_i, b_i) be the intersection of L_{i+} and L_{i-}.
    Let L_{i-} be the bisector of p_j and p_k.
    Prune away p_j (resp. p_k) if p_j (resp. p_k) is closer to (x^*, y^*).
    Case 2 $x_s < x^*$ and $y_s > y^*$ `/* 2nd quadrant */`
    Find all (a_i, b_i)'s such that $a_i > x^*$ and $b_i < y^*$.
    Let (a_i, b_i) be the intersection of L_{i+} and L_{i-}.
    Let L_{i+} be the bisector of p_j and p_k.
    Prune away p_j (resp. p_k) if p_j (resp. p_k) is closer to (x^*, y^*).

Prune and search algorithm (cont'd)

1-center problem

10. Case 3 $x_s < x^*$ and $y_s < y^*$ `/* 3rd quadrant */`
    Find all (a_i, b_i)'s such that $a_i > x^*$ and $b_i > y^*$.
    Let (a_i, b_i) be the intersection of L_{i+} and L_{i-}.
    Let L_{i-} be the bisector of p_j and p_k.
    Prune away p_j (resp. p_k) if p_j (resp. p_k) is closer to (x^*, y^*).
    Case 4 $x_s > x^*$ and $y_s < y^*$ `/* 4th quadrant */`
    Find all (a_i, b_i)'s such that $a_i < x^*$ and $b_i > y^*$.
    Let (a_i, b_i) be the intersection of L_{i+} and L_{i-}.
    Let L_{i+} be the bisector of p_j and p_k.
    Prune away p_j (resp. p_k) if p_j (resp. p_k) is closer to (x^*, y^*).
11. Let S be the remaining points and go to step 1.

Time of prune and search algorithm

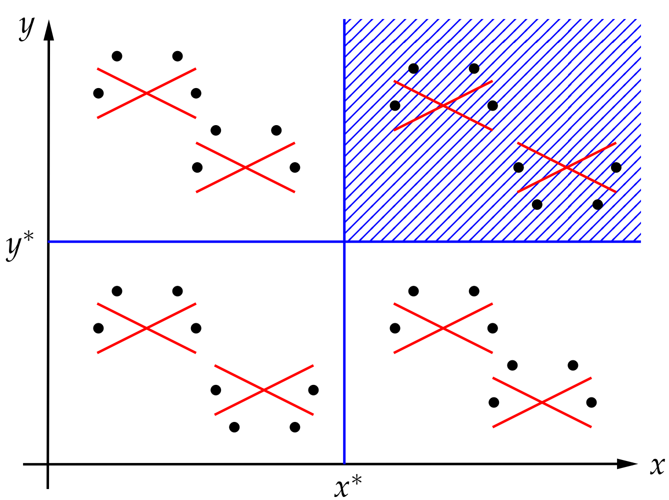
1-center problem

- Assume that there are $n = 16^k$ points for some k.
- There are $\frac{n}{2}$ bisectors formed in step 2.
- After step 4, $\frac{n}{4}$ of them have positive slops and $\frac{n}{4}$ of them have negative slops.
- Hence, there are a total $\frac{n}{4}$ intersections formed in step 5.
- Since x^* (y^*) is the median of a_i's (b_i's), there are $\frac{n}{4} \times \frac{1}{4} = \frac{n}{16}$ (a_i, b_i)'s for each case in step 10.

Time of prune and search algorithm (cont'd)

1-center problem

- Assume that the optimal solution lies in the shaded area.



Time of prune and search algorithm (cont'd)

1-center problem

- It can be verified that for each pair of intersection in the region with $x < x^*$ and $y < y^*$, the point above the line with negative slope can be pruned.
- For each such (a_i, b_i), exactly one point is pruned away.
- Hence, $\frac{n}{16}$ points are pruned away in each iteration.
- Since each iteration takes $\mathcal{O}(n)$ time, the total time complexity of the prune and search algorithm is $\mathcal{O}(n)$.