Astronomer Solution Sketch

David R. Lolck

29. April 2023

Problem

Problem

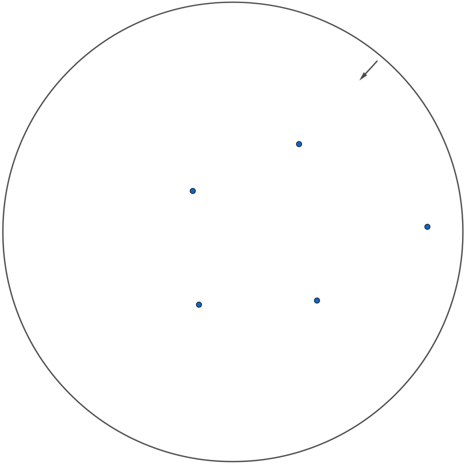
Given n points and integers k, s, t, determine the minimum cost circle with center C and radius r that contain k points where the cost is calculated as:

$$\mathrm{cost}(C, r) = s \cdot d((0,0), C) + t \cdot r$$

Sketch

Observation 1

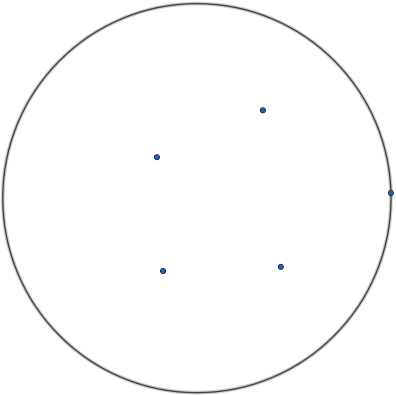
The optimal circle has at least 1 point on the perimeter.



Sketch

Observation 1

The optimal circle has at least 1 point on the perimeter.



Sketch

Subtask 1: $t \leq s$

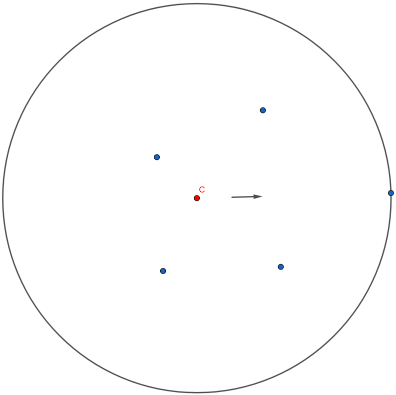
If $t \leq s$, then the solution is the distance to the kth closets point times t. Running time $O(n \lg n)$ for sorting.

We can therefore assume $t > s$ from this point onward. So increasing the circle is more expensive than moving the center.

Sketch

Observation 2

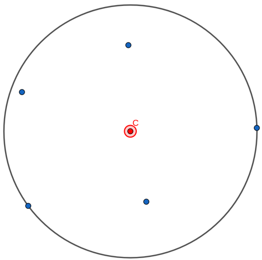
The optimal circle has at least 2 points on the perimeter.



Sketch

Observation 2

The optimal circle has at least 2 points on the perimeter.

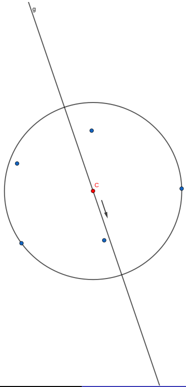


Sketch

Observation 3

The optimal circle either:

- Has at least 3 points on the perimeter, or
- is the minimal cost center on some bisector between two points p and q, such that p and q lies on the perimeter.

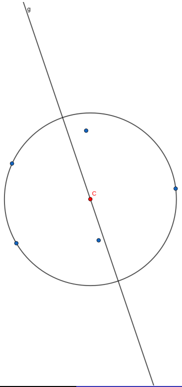


Sketch

Observation 3

The optimal circle either:

- Has at least 3 points on the perimeter, or
- is the minimal cost center on some bisector between two points p and q, such that p and q lies on the perimeter.



Sketch

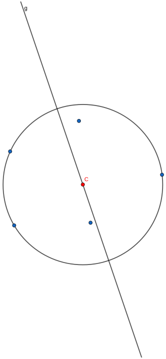
Subtask 2: $n \leq 50, s = 0$

Check all $O(n^3)$ candidates for centers with 3 points, each in $O(n)$ time. Optimal cost between two points lie directly between them. Check all $O(n^2)$ candidates in $O(n)$ time, for total time $O(n^4)$.

Subtask 4: $n \leq 50$

As before, but optimal cost between two points is found through ternary search on bisector. Check all $O(n^2)$ candidates in $O(n + \lg \epsilon^{-1})$ time, for total time $O(n^4)$.

Sketch



Subtask 5: $n \leq 350$

For every bisector between points p and q, for each r of all other points, determine the interval of the bisector where r is contained in a circle with center on the bisector and p and q on the perimeter. Determine all points overlapped by k intervals. Can be done in $O(n^3 \lg n)$ time.

Sketch

Idea for $s = 0$

Fix r. Determine whether there exists a circle with radius r that contain k points. Use this to binary search r.

Sketch

Subtask 3: $s = 0$

Fix r and some point p. Sweep a circle of radius r around p, containing p on the perimeter. For each other point, determine the angle interval where the circle contains p. Determine if there exist k overlapping intervals. Binary search r and iterate p. Running time $O(n^2 \lg \epsilon^{-1} \lg n)$

Sketch

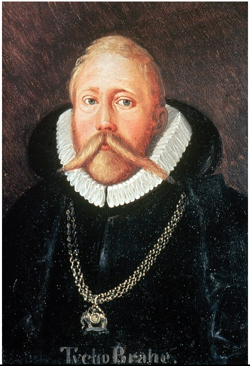
Subtask 6: $\epsilon = 1/10$

Fix c and some point p. Sweep a circle of cost c around p, containing p on the perimeter. For each other point, determine the angle interval where the circle contains p. Determine if there exist k overlapping intervals. Binary search c and iterate p. Gives a running time of $O(n^2 \lg \epsilon^{-2})$.

Sketch

Subtask 6: $\epsilon = 1/10$

Fix c and some point p. Sweep a circle of cost c around p, containing p on the perimeter. For each other point, determine the angle interval where the circle contains p. Determine if there exist k overlapping intervals. Binary search c and iterate p. Gives a running time of $O(n^2 \lg \epsilon^{-2})$.



Sketch

Subtask 6: $\epsilon = 1/10$

Fix c and some point p. Sweep a circle of cost c around p, containing p on the perimeter. For each other point, determine the angle interval where the circle contains p. Determine if there exist k overlapping intervals. Binary search c and iterate p. Gives a running time of $O(n^2 \lg \epsilon^{-2})$.

Subtask 7: *No further constraints*

Same as subtask 6, but observe that we for each point p can determine the best cost that has p on the perimeter and contains k points. They have some ordering. Like starring contest, if we shuffle the points, we only expect to $O(\lg n)$ times observe a lower cost. Gives a running time of $O(n^2 \lg \epsilon^{-1} + n \lg \epsilon^{-2} \lg n)$