Mineral Deposits Solution Sketch

David R. Lolck

April 30, 2023

Problem

Problem

There are k hidden points $(x_1, y_1), \ldots, (x_k, y_k)$.
Access to following query for:

$$Ask((s_1, t_1), \ldots, (s_d, t_d)) = \{|s_i - x_j| + |t_i - y_j| \\ \text{for } (i,j) \in \{1, \ldots, d\} \times \{1, \ldots, k\}\}$$

Determine the hidden points, minimising the number of queries.

Sketch

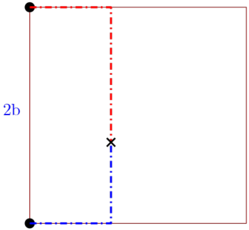
Subtask 1: $k = 1, w = 10^4$

We can with the two queries

$$Ask((-b, -b)) = a$$

$$Ask((-b, b)) = c$$

uniquely determine the single point



Sketch

Observation 1

If we were to ask the queries

$$Ask((-b, -b)) = a_1, \ldots, a_k$$

$$Ask((-b, b)) = c_1, \ldots, c_k$$

Then for each hidden point (x_i, y_i) there exists a pair of distances c_s, d_t, such that those would be returned if (x_i, y_i) was the only hidden point.
Trying all pairs of distances gives k^2 candidate points. This is a superset of the hidden points.

Sketch

Observation 1

If we were to ask the queries

$$Ask((-b,-b)) = a_1, \ldots, a_k$$

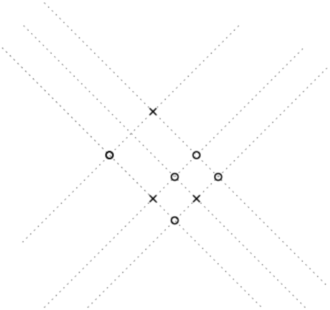
$$Ask((-b,b)) = c_1, \ldots, c_k$$

Then for each hidden point (x_i, y_i) there exists a pair of distances c_s, d_t, such that those would be returned if (x_i, y_i) was the only hidden point.
Trying all pairs of distances gives k^2 candidate points. This is a superset of the hidden points.

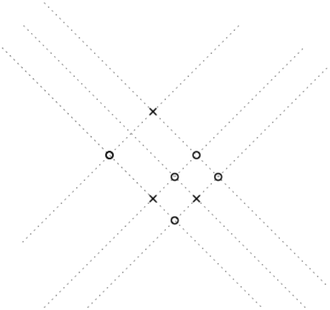
Observation 2

If you ask a query $Ask((s_1, t_1), \ldots, (s_d, t_d))$, then the number of times 0 occurs in the answer corresponds to the number of mineral deposits in $(s_1, t_1), \ldots, (s_d, t_d)$.

Sketch



Sketch



Subtask 2: $w \geq 500$

Reduce to k^2 candidate points. Ask a query for each of them. If the result contains a 0, queried point is a mineral deposit.

Sketch

Subtask 3: $w \geq 210$

Reduce to k^2. We can locate 1 point in $\lceil \lg(k^2) \rceil$ queries. Split k^2 candidates in two sets and query one of them. Then we can determine if there is a mineral deposit in this set or the other one. Once we have located a deposit, never ask about it again.

Sketch

Subtask 3: $w \geq 210$

Reduce to k^2. We can locate 1 point in $\lceil \lg(k^2) \rceil$ queries. Split k^2 candidates in two sets and query one of them. Then we can determine if there is a mineral deposit in this set or the other one. Once we have located a deposit, never ask about it again.

Subtask 4: $w \geq 130$

Same as before, but count the number of mineral deposits in each set, and find all of them at once.

Sketch

Subtask 3: $w \geq 210$

Reduce to k^2. We can locate 1 point in $\lceil \lg(k^2) \rceil$ queries. Split k^2 candidates in two sets and query one of them. Then we can determine if there is a mineral deposit in this set or the other one. Once we have located a deposit, never ask about it again.

Subtask 4: $w \geq 130$

Same as before, but count the number of mineral deposits in each set, and find all of them at once.

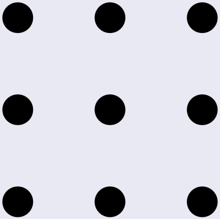
Honorable mention: Random selection

If you simply select a random candidate point, query it, and then remove inconsistent candidate points, this will perform better than the binary search based solutions.

Sketch

Aside

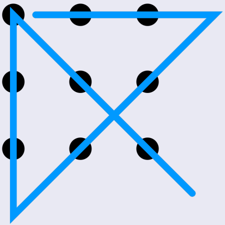
Some of you have seen the following puzzle: Cover all 9 dots with only 4 lines.



Sketch

Aside

Some of you have seen the following puzzle: Cover all 9 dots with only 4 lines.



You have to think outside the box, and we will do the same.

Sketch

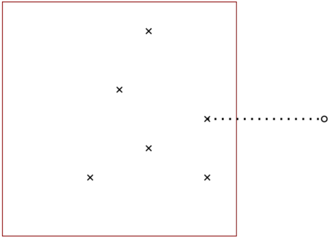
Observation

Let (u, v) be the right-most candidate point. Then placing a point at coordinate $(b + x, v)$ for some x, tells us whether (u, v) is a deposit.

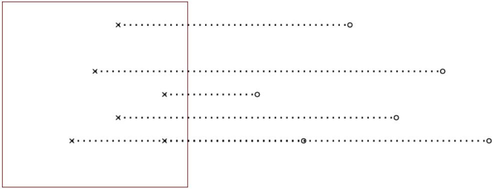
Sketch

Observation

Let (u, v) be the right-most candidate point. Then placing a point at coordinate $(b + x, v)$ for some x, tells us whether (u, v) is a deposit.



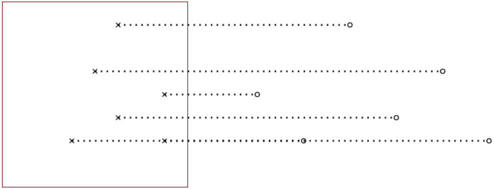
Sketch



Subtask 5: $w \geq 3, b \leq 10^4$

Process candidates from right to left. On the same y coordinate as the candidate, place outside the query point. If the distance between the query points are at least $5b$, then the distance in the response can each be associated with a query point. E.g. the query points become $(x, 5b), (y, 10b), (z, 15b), \ldots$. Query the points

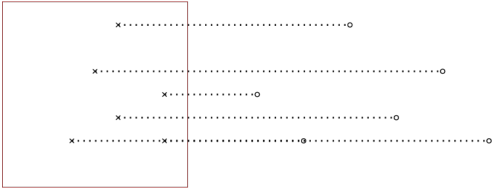
Sketch



Subtask 5: $w \geq 3, b \leq 10^4$

Process candidates from right to left. On the same y coordinate as the candidate, place outside the query point. If the distance between the query points are at least $5b$, then the distance in the response can each be associated with a query point. E.g. the query points become $(x, 5b), (y, 10b), (z, 15b), \ldots$. Query the points For each candidate, if the distance between the associated query point and the candidate is in the result of the query, it is a deposit. Then undo all distances that are a result of this deposit. Uses distances of $O(k^2 b)$ outside the b-box

Sketch



Subtask 6: $w \geq 3, b \leq 10^7$

Same idea as before. Observe that we can instead compute all the k^2 distances that could arise from each query point. Determine the smallest distance that hasn't been generated further away than the previous point and place it there. It can be shown that this uses query points of at most k^4 distance outside the b-box.

Sketch

Observation

We can reduce to $4k^2$ query points using only a single query, by combining the two queries into one and taking every possible combination of distances.

Sketch

Observation

We can reduce to $4k^2$ query points using only a single query, by combining the two queries into one and taking every possible combination of distances.

Subtask 7: *No further restrictions*

Observe that when b is very large, there is going to be a lot of empty space. Do essentially the same as subtask 6, but in all 4 directions at the same time. You have to make sure that the points remain uniquely decodeable while placing new query points. You also have to make sure that new query points are not too close to candidate points, and that you don't duplicate some important distances.
Running time: Depending on implementation either $O(k^4)$ or $O(k^6)$.