Homework Exam 1 2022-2023

Alexander Apers
6272932

Deadline: 25 November 2022, 13:15

This homework exam has 1 question for a total of 9 points. You can earn an additional point by a careful preparation of your hand-in: using a good layout, good spelling, good figures, no sloppy notation, no statements like "The algorithm runs in $n \log n$." (forgetting the $O(..)$ and forgetting to say that it concerns time), etc. Use lemmas, theorems, and figures where appropriate.

Question 1 (9 points)

Let $r \in \mathbb{R}^2$ be a "red" point, and let B be a set of n "blue" points in $\mathbb{R}^2$. You can assume that the points are in general position; meaning that no two points have the same x-coordinate or the same y-coordinate, and that no three points lie on a line. A triangle is "bichromatic" when its vertices are either red or blue, and it has at least one vertex of either color. Develop an $O(n \log n)$ time algorithm to find a maximum area "bichromatic" triangle Δ^* on $\{r\}, B$.

Hint: You can use the following fact. A function $f[1..n] \to \mathbb{R}$ is *unimodal* if (and only if) it has a single (local) maximum. A maximum of f can be computed in $O(T \log n)$ time, where T is the time it takes to evaluate a single value $f(i)$ with $i \in [1..n]$. In particular, using the following function $\textsc{TernarySearch}([1, n], f)$:

function $\textsc{TernarySearch}([a..b], f)$
    $n \leftarrow b - a$
    if $n < 3$ **then** evaluate $f(i)$ for each $i \in [a..b]$ and return $\max_i f(i)$
    else
        $m_1 \leftarrow a + \lfloor n/3 \rfloor$; $m_2 \leftarrow a + \lfloor 2n/3 \rfloor$
        if $f(m_1) < f(m_2)$ **then** $\textsc{TernarySearch}([m_1.., b], f)$
        else$\textsc{TernarySearch}([a..m_2], f)$
        end if
    end if
end function

Solution to Question 1

Definitions

Let A denote the set $\{r\} \cup B$
Let $\mathcal{CH}(A)$ denote the convex hull of point set A (including interior and edges)
Let CH(A) denote the set of vertices that make up the edges of $\mathcal{CH}(A)$

Geometric Observations

Since there is only one red point, it is clear that all bichromatic triangles contain exactly one red point and two blue points. We start with the following observation.

Lemma 1. *The maximum area bichromatic triangle must necessarily consist of 2 blue vertices, b_1, b_2 such that $b_1, b_2 \in$ CH(A).*

Proof. Suppose for contradiction that $b_1 \notin$ CH(A). We denote the maximum area triangle $\triangle rb_1b_2$. Imagine the line l parallel to the line going through r and b_2 which goes through the point b_1. If we replace b_1 with any point on the other side of l compared to where r and b_2 are, the area of $\triangle rb_1b_2$ will increase as this point lies farther from the segment $\overline{rb_2}$ than b_1. l must cross the boundary of $\mathcal{CH}(A)$ in 2 locations since it goes through point $b_1 \in \mathcal{CH}(A)$ but $b_1 \notin$ CH(A). Therefore, there must be a least some point $x \in$ CH(A) further away from $\overline{rb_2}$ than b_1 to connect the two points where l crosses $\mathcal{CH}(A)$. Therefore, it's guaranteed that b_1 can be replaced by at least one point $x \in$ CH(A) so that the area of $\triangle rxb_2 > \triangle rb_1b_2$. Hence, $\triangle rb_1b_2$ is not a maxmium area triangle. This is a contradiction, since we assumed $\triangle rb_1b_2$ had maximum area. We conclude that $b_1 \in$ CH(A) to be a maximum area triangle. □

Similarly, it can be proven that $b_2 \in$ CH(A).

Lemma 2. *Given point r and a point $b_1 \in$ CH(A), and line l going through them, TriangleArea($\triangle rb_1b_2$) is a unimodal function for $b_2 \in$ CH(A) on one side of l.*

Proof. The area of $\triangle rb_1b_2$ is defined as $d(r, b_1) \cdot d(b_2, \overline{rb_1})$. The only input to the function which changes is the point b_2. Therefore, the area of the triangle is determined by b_2. If the set of vertices in CH(A) on one side of l is not empty, the distances $d(b_2, \overline{rb_1})$ have to be (1) monotonically non-decreasing, then (2) maximal, and finally (3) monotonically non-increasing, in that order when iterating through them in clockwise order. If this is not the case you can find 3 consecutive points $x, y, z \in$ CH(A) such that $d(y, \overline{rb_1}) < d(x, \overline{rb_1})$ and $d(y, \overline{rb_1}) < d(z, \overline{rb_1})$. If this were the case then CH(A) would contain vertices that create a non-convex polygon. Therefore, TriangleArea($\triangle rb_1b_2$) is a unimodal function for $b_2 \in$ CH(A) on one side of l. □

Algorithm Description

Finding the vertices that make up the boundary of the convex hull can be done in $O(n \log n)$ time complexity using the algorithm discussed in class. These vertices will be given in clockwise (or counterclockwise) order.

The algorithm iterates through all points $b_1 \in$ CH(A). For every point, the line l is determined that goes through r and b_1. Then, CH(A) is partitioned into a subset with points left of l and a subset with points right of l. The consecutive points i, j such that i and j are on opposite sides can be found using a simple BinarySearch. At every step, half of the input can be eliminated by evaluating which side of the line l the middle point in the input is on. Since the AreaTriangle function is unimodal on both sides of the line, we apply TernarySearch once for both sides of l. The largest of these two triangles is the maximum area triangle that uses the vertices b_1 and r. Since this is done for all $b_1 \in$ CH(A), we simply have to keep track of the largest area triangle, which is returned at the end.

function $\textsc{MaxAreaBichromaticTriangle}(r, B, \text{CH}(A))$
  MaxArea $\leftarrow 0$
  MaxAreaTriangle $\leftarrow \triangle$
  for $k \leftarrow 1$ to m **do**
    $b_2 \leftarrow \text{CH}(A)[k]$
    $l \leftarrow$ line through r and b_1
    $i, j \leftarrow$ BinarySearch(CH(A), l)
    left $\leftarrow$ subset of CH(A) left of l
    right $\leftarrow$ subset of CH(A) right of l
    $b_{2\text{left}} \leftarrow$ TernarySearch(left, AreaTriangle)
    $b_{2\text{right}} \leftarrow$ TernarySearch(right, AreaTriangle)
    $b_{\text{max}} \leftarrow \max(b_{2\text{left}}, b_{2\text{right}})$
    if AreaTriangle($\triangle r b_1 b_{\text{max}}$) $\geq$ MaxArea **then**
      MaxArea $\leftarrow$ AreaTriangle($\triangle r b_1 b_{\text{max}}$)
      MaxAreaTriangle $\leftarrow \triangle r b_1 b_{\text{max}}$
    end if
  end for
end function

Algorithm Correctness

It is guaranteerd that the largest area bichromatic triangle is discovered by the algorithm. This is because the two TernarySearch calls find the vertex b_2 such that $\triangle r b_1 b_2$ has maximum area for every $b_1 \in \text{CH}(A)$. TernarySearch can be applied using the AreaTriangle function and the given input due to **Lemma 2**. **Lemma 1.** shows that the two blue points need to be on the edge of the convex hull to be the maximum area triangle.

Running Time Analysis

The outer loop loops over every vertex $b_1 \in \text{CH}(A)$. There is an $O(n)$ number of vertices in CH(A). Binary search to find the two consecutive vertices in CH(A) that are on opposite sides of l can be done in $O(\log n)$. Applying TernarySearch twice on points in CH(A) on either side of l can be done in $O(\log n)$ time complexity, since evaluating AreaTriangle (calculating the area of a triangle with given vertices) can be done in $O(1)$ time complexity. This means that the whole algorithm will run in $O(n \log n)$ time complexity.