Closest Segments

Kathryn Duffy* Chris McAloney* Henk Meijer* David Rappaport*

Abstract

We present an algorithm that determines for every point $r \in P$ the closest distance between r and a line segment (p, q) whose end points are from $P \backslash \{r\}$. Our algorithm can be implemented in $O(n^2)$ time and $O(n)$ space. Since we show that our problem is 3SUM-hard it is unlikely that a faster algorithm will be found soon.

1 Introduction

In this paper we address proximity problems for a set of points in the plane and the set of line segments determined by these points.

Overall closest line segment and point. Given a set S of n points in the plane, determine the point and line segment pair which has the smallest proximity over all possible pairs; *i.e.* determine three distinct points $p, q, r \in S$ such that the distance between the point p and the line segment (q, r) is the smallest.

All-points closest line segment. Given a set S of n points in the plane, report for each point $p \in S$ the closest line segment with endpoints in $S \backslash \{p\}$; *i.e.* for each point $p \in S$, find the two distinct points $q, r \in S \backslash \{p\}$ such that the distance between the point p and the line segment (q, r) is the smallest.

These problems were inspired from the work of Daescu and Luo [1] which demonstrated that given a set S of n points in the plane and a point $p \in S$, a line segment with endpoints in $S \backslash \{p\}$ that is closest to p can be found in $O(n \log n)$ time and $O(n)$ space. As there are $O(n^2)$ line segments to consider, their algorithm is a substantial improvement over the brute force method.

A direct application of Daescu and Luo's algorithm would result in an $O(n^2 \log n)$ algorithm for both proximity problems. In contrast, we give a different approach to the all-points closest line segment problem which results in an $O(n^2)$ algorithm requiring $O(n)$ space. Trivially, this algorithm also solves the overall closest line segment and point problem. We provide evidence that the time complexity would be hard to improve upon by showing that these problems are 3SUM-hard.

*School of Computing, Queen's University, {duffy, mcaloney, henk, daver} @cs.queensu.ca

2 Closest Segments

Let $P = \{p_0, p_1, \ldots, p_{n-1}\}$ be a set of n distinct points in the plane. A line segment is defined by two points in P and is denoted by (p_i, p_j). Let $d(p_i, p_j)$ denote the Euclidean distance between points p_i and p_j. Let $d(p_i, e)$ denote the Euclidean distance between point p_i and line segment e. Let $c(p_i)$ denote the point in $P \backslash \{p_i\}$ that is closest to p_i. In case of ties, $c(p_i)$ denotes one of the points closest to p_i. Let $e(p_i)$ denote one of the line segments (p_j, p_k) with $i \neq j \neq k \neq i$ that is closest to p_i. We say that a line segment e is a relevant closest line segment of p_i if $d(p_i, e) = d(p_i, e(p_i))$ and if $d(p_i, e) < d(p_i, c(p_i))$. Let $C(p_i, p_j)$ be the closed disk, such that p_i and p_j lie on its boundary and such that its diameter has length $d(p_i, p_j)$.

We first observe the following: Let p denote a point in P. Let $q_0, q_1, \ldots$ be some of the remaining $n-1$ points, numbered in counterclockwise radial order around p. Let $\alpha_{i,j}$ denote the counterclockwise angle between edges (p, q_i) and (p, q_j). We observe that if $\pi/2 \leq \alpha_{0,i} \leq 3\pi/2$ or $q_i \in C(p, q_0)$ then (p, q_i) is not a relevant closest line segment of q_0.

Lemma 1 *Assume $q_{i+1} \in C(p, q_i)$ for $i = 0, 1, \ldots, k-2$ with $k > 1$ and $\alpha_{0,k-1} < \pi/2$. Then (p, q_{k-1}) is not a relevant closest line segment of q_0.*

Proof. For an illustration, see Figure 1 where $k = 4$. If $q_{k-1} \in C(p, q_0)$ then (p, q_{k-1}) is not a relevant closest line segment of q_0, so we may assume that $k > 2$ and $q_{k-1} \notin C(p, q_0)$.

Without loss of generality assume that p lies at the origin and q_0 on the positive x-axis. Let A be the part of the circle centered at q_0 and passing through q_{k-2} that lies above the line through q_0 and q_{k-2}. Consider the angle β between the two halflines anchored at q_{k-2}, tangent to A and $C(p, q_{k-2})$, where the latter tangent is the one that

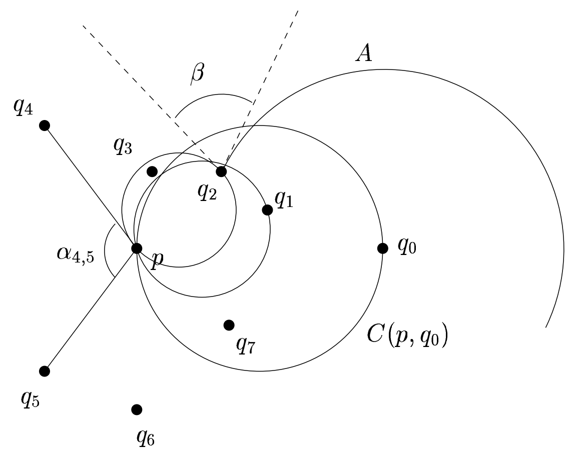

Figure 1: Points $q_{i+1} \in C(p, q_i)$ for $i = 0, 1, 2$.

lies above q_{k-2}. Since the angle between (p, q_{k-2}) and (q_{k-2}, q_0) lies between 0 and π, it follows that $0 < \beta < \pi$. Therefore the arcs A and $C(p, q_{k-2})$ do not intersect except at q_{k-2}. So $(p, q_{k-2}) \cup A$ forms a simple curve that separates q_0 from all points in $C(p, q_{k-2})$ that lie above (p, q_{k-2}). Let q be the point on (p, q_{k-1}) that is closest to q_0. The line segment (q_0, q) either intersects (p, q_{k-2}) or A. Since all points on A have a distance to q_0 that is equal to $d(q_0, q_{k-2})$, we have $d(q_0, q) > d(q_0, (p, q_{k-2}))$ or $d(q_0, q) > d(q_0, q_{k-2})$. So the lemma follows. □

Let p denote a point in P. Let $q_0, q_1, \ldots, q_{n-2}$ be the remaining $n-1$ points, numbered in counterclockwise radial order around p. If two points have the same radial order, we first number the point that is closest to p. In the remainder of this paper, we assume that arithmetic in the indices of points q_i, $0 \le i < n-1$ is defined modulo $n-1$. Suppose that (p, q_{i+j}) is a relevant closest line segment of some point q_i. As before, let $\alpha_{i,i+j}$ denote the counterclockwise angle at p between (p, q_i) and (p, q_{i+j}). From the definition of a relevant closest line segment we know that $-\pi/2 < \alpha_{i,i+j} < \pi/2$. We will show that if $0 \le \alpha_{i,i+j} < \pi/2$ then for all h with $0 \le h < j$ we have that q_{i+j} lies outside $C(p, q_{i+h})$.

Lemma 2 *Let $P = \{p, q_0, q_1, \ldots, q_{n-2}\}$ be a set of n distinct points in the plane such that $\{q_0, q_1, \ldots, q_{n-2}\}$ are numbered in counterclockwise order around p. Assume that the segment (p, q_{i+j}) is a relevant closest line segment of the point q_i with $0 \le \alpha_{i,i+j} < \pi/2$. Then q_{i+j} is outside $C(p, q_{i+h})$ for any point q_{i+h} with $0 \le h < j$.*

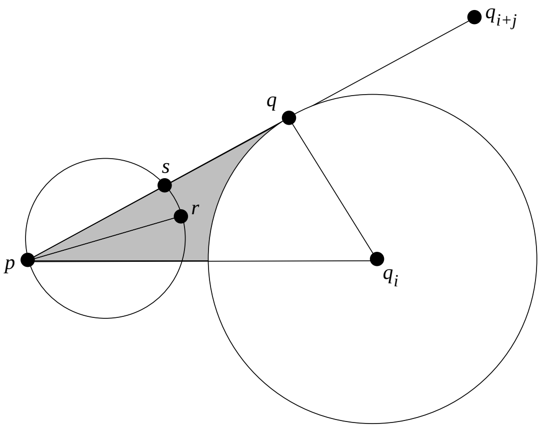

Figure 2: Illustration of the proof of Lemma 2.

Proof. If $j = 1$ the lemma follows immediately from the definition of relevant closest line segments. So we may assume that there are points q_{i+h} with $0 < h < j$. Let q be the point on (p, q_{i+j}) closest to q_i and consider the region, R, between the triangle pqq_i and the circle centered at q_i and passing through q (this is the shaded region in Fig. 2). Suppose that some point q_{i+h} with $0 < h < j$ lies outside R. If q_{i+h} lies in the circle centered at q_i and passing through q then $d(q_i, q_{i+h}) < d(q_i, (p, q_{i+j}))$, which contradicts the assumption that (p, q_{i+j}) is a relevant closest line segment of q_i. If q_{i+h} lies outside this circle, then $d(q_i, (p, q_{i+h})) < d(q_i, (p, q_{i+j}))$, which also contradicts the assumption that (p, q_{i+j}) is a relevant closest line segment of q_i. Therefore, all points $q_{i+1}, \ldots, q_{i+j-1}$ must fall within R.

Since the triangle pqq_i has a right angle at q we note that the triangle pqq_i is completely contained in $C(p, q_i)$.

Let r be any point in the region R, and let $s = C(p, r) \cap (p, q) \setminus \{p\}$. Since (p, r) is a diameter of $C(p, r)$ and (p, s) is a chord of $C(p, r)$ not passing through r, we have that $d(p, s) < d(p, r)$. Let $\mathcal{C}_q$ be the circle centered at p and passing through q. Since $R \subset \mathcal{C}_q$, we also have that $d(p, r) \le d(p, q)$. Therefore $q_{i+j} \notin C(p, r)$ for any point $r \in R$. □

We now implement the ideas of the previous two lemmas with a simple iterative algorithm.

We obtain for each point p in P an angular sorting. Letting point p be the origin of a coordinate system. The angle $\beta(q)$ of a point $q \neq p$ is defined as the counterclockwise angle formed by the positive x-axis and the line segment (p, q). If two or more points result in the same angle we break ties using the distance from p, that is, points closer to p

precede those that are further away. Observe that it suffices to sweep in one direction. We would need to sweep the points once around and a bit more, $5\pi/2$ radians to be precise. However, to simplify the description of the algorithm when handling wrap-around in a circular sequence, we double the length of the angularly sorted list so that the second appearance of a point q is assigned an angle $\beta(q) + 2\pi$ yielding a sequence of points $(q_0, q_1, \ldots q_{2n-3})$.

One can obtain all n angle sorted sequences, one for each point p in P, in $O(n^2)$ time and $O(n)$ space by using a so called topological sweep of the line arrangement of the n lines obtained from a standard point line dual of P, see [2]. A simpler method for obtaining these sequences in $O(n^2)$ time and $O(n)$ space, without using the dual transform, can be found in [5].

For each point $r \in P$ we use Best(r) to maintain the value of a currently known closest line segment. We can initialize Best(r) for each point r with the distance to one of its nearest neighbors $c(r)$. It is well known that all nearest neighbours for a set of n points can be obtained in $O(n \log n)$ time [4, p. 184]. If need be we could also store together with the value Best(r) a witness, either a point or a line segment that realizes the distance Best(r) with r.

A simplified high level description of the algorithm precedes the detailed pseudo code presentation. We use the approach suggested by our Lemmas and maintain an active list of vertices using a double ended queue, also known as a deque, Q. We use first(Q) and last(Q) to denote the two ends of the deque. If the deque is empty first(Q) and last(Q) return a null value. Whenever a point q is found such that q is outside of the circle $C(p, r)$ we determine that (p, q) is a closest relevant segment to r using endpoint p. The detailed algorithm follows.

```
Algorithm Relevant
insertlast (q_0, Q)
for i = 1 to 2n − 3
  q_j ← first(Q)
  while ( Q is not empty and α_ij ≥ π/2 ) do
    removefirst(Q)
    q_j ← first(Q)
  end while
  q_j ← last(Q)
  while (  Q is not empty and q_i ∉ C(p, q_j) ) do
    Best(q_j) ← min(Best(q_j), d(q_j, (p, q_i)))
    removelast(Q)
    q_j ← last(Q)
  end while
  insertlast(q_i, Q)
end for
```

We maintain the invariant that points on Q are exactly those points referred to in the Lemmas. That is, without loss of generality there is a labelling of points such that $q_0, q_1, ..., q_{k-1}$ are on the points on Q $q_{i+1} \in C(p, q_i)$ for $i = 0, 1, \ldots, k-2$ with $k > 1$ and $\alpha_{0,k-1} < \pi/2$. Thus correctness follows immediately.

The running time of the Algorithm Relevant is characterized by its for loop. Although we employ two while loops it is easy to see that the time spent in the while loops for the entire life of the algorithm is bounded by $2n$, because any instance of a point in the angular order is inserted to and/or removed from Q at most once.

Thus, the cost of obtaining the closest segment for every point $p \in P$ is $O(n^2)$.

We show that our problem belongs to the class 3SUM-hard as defined by Gajentaan and Overmars [3]. Problems that are 3SUM-hard are reducible from 3SUM, a problem that has eluded an $o(n^2)$ algorithm despite concentrated effort. Given three sets of integers, A, B, and C with total size n, the 3SUM problem asks to determine whether there are elements $a \in A$, $b \in B$, and $c \in C$ such that $a + b = c$. The problem to determine whether 3 points are collinear in an input of n points, 3-points-on-line, is shown to be 3SUM-hard [3]. Observe that the problems discussed in this paper solve 3-points-on-line since the existence of three points on a line results in the overall closest line segment and point. This remark is formalized in the following lemma.

Lemma 3 *Both the overall closest line segment and point problem and the all-points closest line segment problem are 3SUM-hard.*

References

[1] Ovidiu Daescu and Jun Luo Proximity Problems on Line Segments Spanned by Points, *Proc. 14th Annual Fall Workshop on Computational Geometry*,MIT, Cambridge, Mass., USA, 9-10, 2004

[2] H. Edelsbrunner and L.J. Guibas Topologically sweeping an arrangement, . *J. Comput. Syst. Sci.*, 38:165-194, 1989. Corrigendum in 42:249-251, 991.

[3] A. Gajentaan and M. H. Overmars On a class of $O(n^2)$ problems in computational geometry, *Comput. Geom. Theory Appl.*, 5:165-185, 1995.

[4] J. O'Rourke*Computational Geometry in C*, Cambridge University Press, 1994.

[5] M. H. Overmars and E. Welzl, New methods for computing visibility graphs, *Proc. 4th Annu. ACM Sympos. Comput. Geom.*, 164-171, 1988