On 1-Segment Center Problem

H. Imai
Kyushu University, Fukuoka 812, Japan

D. T. Lee\(^1\) and C. D. Yang
Northwestern University, Evanston, IL

Abstract

The location problem has recently been investigated from the computational-geometric point of view. The most fundamental problem is the 1-point-center problem, or the minimum enclosing circle problem, for \(n\) demand points in the plane, which is to find a location of a point facility \(p\) so that the maximum distance from \(p\) to the demand points is minimized. Megiddo and Dyer presented an \(O(n)\) optimal time algorithm for this problem. A variation of the 1-point center problem is the 1-line-center problem, which calls for the location of a line facility so that the maximum distance from the \(n\) given points to the line is minimized. This problem can be solved in \(\Theta(n \log n)\) time, which is shown to be optimal in the worst case under the algebraic computation tree model of Ben-Or.

The complexities of the 1-point-center and 1-line-center problems are essentially different, and thus naturally arises the following problem, called the 1-segment-center problem: Given a set \(S\) of \(n\) points in the plane and a nonnegative constant \(L\), locate a line segment of length \(L\) so that the maximum distance between the segment and the points in \(S\) is minimized. The distance between a point \(p\) and a segment \(l\) is the minimum distance between \(p\) and any point on \(l\). The placement of a segment can be represented by \((x, y, \theta)\), where \((x, y)\) are the coordinates of one (designated) endpoint of the segment, and \(\theta\) is the orientation of that segment with respect to the \(X\)-axis. Given a segment \(l\) of length \(L\) placed at \(\hat{p} = (x, y, \theta)\), the locus of points equidistant from \(l\) at distance \(r\) is called a segment disk centered at \(\hat{p}\) of radius \(r\), and is denoted as \(l(L, \hat{p}, r)\).

It can be shown that there exist at most four points that determine the location of the segment and hence \(O(n^4 \log n)\) time suffices to find such a disk. However, for some restricted cases, more efficient algorithms can be obtained. For example, if the orientation \(\theta\) is fixed, the problem can be solved in \(O(n)\) time by prune-and-search technique.

\(^1\)Supported in part by the National Science Foundation under the Grand DCR 8420814.