On Intersection Searching Problems Involving Curved Objects

Prosenjit Gupta ¹  Ravi Janardan ¹  Michiel Smid ²

Abstract

Efficient solutions are given for the problem of preprocessing a set of linear or curved geometric objects (e.g., lines, line segments, circles, circular arcs, d-balls, d-spheres) such that the ones that are intersected by a curved query object can be reported (or counted) quickly. The problem is considered both in the standard setting (where one is interested in all the objects intersected) and in a generalized setting (where the input objects come aggregated in disjoint groups and one is interested in the disjoint groups that are intersected). The solutions are based on geometric transformations, simplex compositions, persistence, and, for the generalized problem, on a method to progressively eliminate groups that cannot possibly be intersected.

1 Introduction

In a generic instance of an intersection searching problem a set, S, of n geometric objects must be preprocessed, so that the k objects of S that are intersected by a query object q can be reported (or counted) efficiently. Examples of S and q are points, lines, line segments, rays, hyperplanes, and simplices. These problems are rich in applications and space and query-time-efficient solutions are known for many of them [Ede87].

Most previous work on these problems assumes that the input objects and the query object are linear or piecewise-linear. To our knowledge, the case where the input and/or the query are curved has been investigated systematically only in [AvKO93, KOA90, Sha91, AM92]. In [AvKO93], searching on curved objects (e.g., circles, disks, circular arcs, Jordan arcs) with linear query objects (e.g., lines, line segments, halfspaces, rays) is considered. In [KOA90] (resp. [Sha91]), the input is a set of disks and the query is a line or line segment (resp. a point).

¹Department of Computer Science, University of Minnesota, Minneapolis, MN 55455, U.S.A. Email: {pgupta, janardan}@cs.umn.edu. The research of these authors was supported in part by NSF grant CCR-92-00270.

²Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany. Email: michiel@mpi-sb.mpg.de. This author was supported by the ESPRIT Basic Research Actions Program, under contract No. 7141 (project ALCOM II).
In [AM92] range searching on point sets in $R^d$ with ranges defined by a constant number of bounded-degree polynomials is considered.

1.1 Overview of results

In this paper, we make further contributions to intersection searching in the curved setting by presenting efficient solutions to three broad classes of problems. For the first two classes below, we will discuss only the reporting version since the counting problems can be solved similarly.

The first thrust of the paper is the design of efficient solutions to the following general problem: "Preprocess a set $S$ of $n$ linear objects so that the $k$ objects that are intersected by a curved query object can be reported efficiently." Thus, this part of our work complements the results in [AvKO93, KOA90, Sha91]. Table 1 summarizes our results.

The second thrust of the paper is the following problem: "Preprocess a set $S$ of $n$ curved objects so that the ones that are intersected by a curved query object can be reported efficiently." This problem was left open in [AvKO93]. Table 2 summarizes our results.

Finally, we consider a generalization of the preceding two classes: Here $S$ consists of $n$ linear or curved objects and the objects come aggregated in disjoint groups. If we assign each object a color, according to the group it belongs to, then our goal is to report the distinct colors of the objects intersected by a query (rather than reporting all the intersected objects as in the case of the standard problem). Such generalized intersection searching problems are rich in applications and have been considered recently in [JL93, GJS93b, AvK93, GJS94] in the context of linear input and query objects. The challenge in these problems is to obtain solutions whose query times are sensitive to the output size, namely the number, $i$, of distinct colors intersected (not the number, $k$, of intersected objects, which can be much larger). Typically, we seek query times of the form $O(f(n) + i \cdot g(n))$, where $f(n)$ and $g(n)$ are polylogarithmic. We also consider the counting version, where we seek polylogarithmic query time. Tables 3 and 4 summarize our results.

1.2 Overview of techniques and contributions

Our results are based on three main approaches. The first approach transforms the curved problem at hand to a simplex range searching problem and solves the latter by suitably composing together known techniques such as partition trees, cutting trees, and spanning paths of low stabbing number. This general approach has been used in [AvKO93]. What makes this part of our work interesting is that the characterization of intersection and the appropriate transform(s) to use

---

3In this paper, the term "curved" means circular or circle-like objects such as circles, disks, circular arcs, annuli, $d$-balls, and $d$-spheres. A $d$-sphere is the boundary of a closed $d$-ball.

4Throughout, $\epsilon > 0$ is an arbitrarily small constant. Whenever $\epsilon$ appears in a query time (resp. space) bound, the corresponding space (resp. query time) bound contains a multiplicative factor which goes to $\infty$ as $\epsilon \to 0$. 