Maintaining the Extent of a Moving Point Set

Pankaj K. Agarwal¹*, Leonidas J. Guibas²**, John Hershberger³, and Eric Veach⁴

¹ Department of Computer Science, Duke University, Durham, NC 27708, USA
² Computer Science Department, Stanford University, Stanford, CA 94305, USA
³ Mentor Graphics Corp., 1001 Ridder Park Drive, San Jose, CA 95131, USA

Abstract. Let S be a set of n moving points in the plane. We give new efficient and compact kinetic data structures for maintaining the diameter, width, and smallest area or perimeter bounding rectangle of the points. When the points in S move with pseudo-algebraic motions, these structures process \( O(n^{2+\epsilon}) \) events. We also give constructions showing that \( \Omega(n^2) \) combinatorial changes are possible in these extent functions even when the points move on straight lines with constant velocities. We give a similar construction and upper bound for the convex hull, improving known results.

1 Introduction

Suppose S is a set of n moving points in the plane. In this paper we investigate how to maintain various descriptors of the extent of the point set, such as diameter, width, smallest enclosing rectangle, etc. These extent measures give an indication of how spread out the point set S is and are useful in various virtual reality applications such as clipping, collision checking, etc. As the points move continuously, the extent measure of interest (e.g., diameter) changes continuously as well, though its combinatorial realization (e.g., the pair of points defining the diameter) only changes at certain discrete times. Our approach is to focus on these discrete changes or events and track through time the combinatorial description of the extent measure of interest.

We do so within the framework of kinetic data structures (KDSs for short), as developed by Basch, Guibas, and Hershberger [3] and further elaborated in Section 2. There are two notable and novel aspects of that framework. Firstly, while extensive work has been done on dynamic data structures in computational geometry [4, 5], this is all focused on handling insertions/deletions of objects and not continuous change. Kinetic data structures by contrast gain

* Support was provided by National Science Foundation research grant CCR-93-01259, by Army Research Office MURI grant DAAH04-96-1-0013, by a Sloan fellowship, by a National Science Foundation NYI award and matching funds from Xerox Corp, and by a grant from the U.S.-Israeli Binational Science Foundation.

** Support was provided in part by National Science Foundation grant CCR-9623851 and by US Army MURI grant 5-23542-A.
their efficiency by exploiting the continuity or coherence in the way the system state changes. Secondly, unlike Atallah’s dynamic computational geometry framework [2], which was introduced to estimate the maximum number of combinatorial changes in a geometric configuration under predetermined motions in a certain class, the KDS framework is fully on-line and allows each object to change its motion at will, due to interactions with other moving objects, the environment, etc.

Section 3 presents new kinetic algorithms for diameter, width, and smallest enclosing rectangle in both the area and perimeter senses. If we assume that the points of S follow pseudo-algebraic motions (defined below), then the number of events processed by each of our algorithms is $O(n^{2+\epsilon})$ (for all $\epsilon > 0$). In particular these bounds prove that none of the extent measures mentioned can change combinatorially more than $O(n^{2+\epsilon})$ times. A quadratic bound is natural for diameter, as it is defined by two points of the set S, but it is somewhat surprising for the other measures, as width is defined by three points, and the minimum bounding rectangles by four or five of the points. The data structures we give are efficient and compact in the KDS sense, though not local.

Section 4 is devoted to giving lower bound constructions for these extent measures under linear point motions: we show that diameter, width, and the two flavors of smallest bounding rectangle can all change $\Omega(n^2)$ times as the $n$ points of S move on straight line trajectories with constant velocities (possibly different for each point). Such lower bound constructions are much easier if we allow quadratic or other higher degree motions—the fact that the same bounds hold with linear motions is quite interesting. Our constructions employ a key component consisting of cocircular (or nearly cocircular) points that move on straight lines while maintaining their (near-)cocircularity. Finally in Section 5 we give a similar construction showing that the convex hull of $n$ points moving linearly in the plane can also change $\Omega(n^2)$ times. We also prove a tighter upper bound than was previously known for the number of combinatorial changes to the convex hull. This bound is $O(n\lambda_s(n))$, where $\lambda_s(n)$ is the length of a Davenport-Schinzel sequence [7], and the parameter $s$ bounds the number of times three points can become collinear. The bound specializes to $O(n^2)$ for linearly moving points—which is therefore tight.

2 Kinetic data structure preliminaries

A kinetic data structure maintains a configuration function of continuously moving data (e.g., diameter, width, etc., of moving points). It does so by maintaining a set of certificates that jointly imply the correctness of the computed configuration function. Each certificate is a geometric predicate on a constant number of data elements, such as, for example, “points A and B are farther apart than points C and D.” The certificates are typically derived from a static algorithm for computing the configuration function. For example, the certificates for maintaining the diameter might include a set of distance comparisons establishing a partial order on the relevant pairwise distances, with a single maximum element.