A Simple Method for Resolving Degeneracies in Delaunay Triangulations

Michael B. Dillencourt* and Warren D. Smith†

Abstract. We characterize the conditions under which completing a Delaunay tessellation produces a configuration which is a nondegenerate Delaunay triangulation of an arbitrarily small perturbation of the original sites. One consequence of this result is a simple method for resolving degeneracies in Delaunay triangulations that does not require symbolic perturbation of the data.

1 Introduction

A data-induced degeneracy (or simply degeneracy) in a geometric computation is a subset of the input that does not satisfy the "general position" assumptions appropriate for the computation. For example, a degeneracy in a line arrangement is a set of three or more concurrent lines. In the context of planar Delaunay triangulations, a degeneracy is either (1) a set of 4 or more cocircular generating sites such that the circle through the sites contains no other generating site in its interior, or (2) a set of three or more collinear generating sites on the boundary of the convex hull.

Handling degeneracies correctly is an important, and subtle, practical issue that arises in the implementation of geometric algorithms. It is generally desirable to resolve a data-induced degeneracy by computing a nondegenerate output that can be realized by an arbitrarily small perturbation of the input. General techniques, based on symbolic perturbation schemes, are developed in [10, 11, 24]. All of these techniques involve considerable computational overhead.

In this paper, we consider the special case of two-dimensional Delaunay triangulations. It is well-known that not all possible triangulations have combinatorially equivalent realizations as Delaunay triangulations [7, 13]. Indeed, only an exponentially small fraction of triangulations have such a realization [23]. Hence one would expect that some care is necessary when removing degeneracies from Delaunay triangulations and indeed, this turns out to be the case. However, the amount of care required turns out to be modest. In particular, we show that general symbolic-perturbation schemes are unnecessary, and that a much simpler method of resolving degeneracies suffices.

Our main result (Theorem 3.1) is a characterization of the conditions under which completing a degenerate Delaunay tessellation yields a configuration which

* Information and Computer Science Department, University of California, Irvine, CA 92717. The support of a UCI Faculty Research Grant is gratefully acknowledged.
† NEC Research Institute, 4 Independence Way, Princeton NJ 08540
is the nondegenerate triangulation of an arbitrarily small perturbation of the input. The proof of Theorem 3.1 is given in Section 5. The proof is based on a characterization of the conditions under which adding an edge to an inscribable polyhedron preserves inscribability (Theorem 5.1), which may be of independent interest. A specific example showing that the (minor) restrictions imposed by Theorem 3.1 are indeed necessary is given in Section 6. Practical consequences for Delaunay triangulation algorithms are discussed in Section 7. In Section 8, we apply Theorem 3.1 to show that any triangulation of a simple polygon may be realized as a Delaunay triangulation of an "almost cocircular" set of points.

Edelsbrunner has defined a globally equiangular triangulation to be a triangulation that maximizes lexicographically the angle sequence (sorted in increasing order) [9]. Mount and Saalfeld have given an efficient algorithm to compute the globally equiangular triangulation of a set of cocircular points [16]. When points are in general position, the Delaunay triangulation is the globally equiangular triangulation. In Section 9, we show that if a set of points has a degenerate Delaunay tessellation, the globally equiangular triangulation need not be realizable as the nondegenerate Delaunay triangulation of a small perturbation of the sites, even if the globally equiangular triangulation is unique.

Somewhat related to Theorem 3.1 is a recent algorithm by Fortune for computing "approximate" Delaunay triangulations using fixed-precision arithmetic [12]. Fortune's algorithm uses $O(n^2)$ fixed-precision operations and produces a triangulation that satisfies an approximate Delaunay condition. However, there is no guarantee that the output of his algorithm will be the Delaunay triangulation of any input. Our theorem, like the general schemes of [10, 11, 24], is only applicable if exact arithmetic is being used (since otherwise it is impossible to correctly detect degeneracies.) If exact arithmetic is used, our theorem provides a means of ensuring that the output is a Delaunay triangulation of an arbitrarily small perturbation of the input.

2 Preliminaries

Except as noted, we use the graph-theoretical notation and definitions of [2]. In particular, $V(G)$ and $E(G)$ denote the set of vertices and edges of a graph $G$, respectively. A triangulation is a 2-connected plane graph in which all faces except possibly the outer face are bounded by triangles. A maximal planar graph is a planar graph in which all faces (including the outer face) are bounded by triangles. A graph $G$ is 1-tough [4] if for all nonempty $S \subseteq V(G)$, $c(G - S) \leq |S|$. (Here $|\cdot|$ denotes cardinality, and $c(\cdot)$ denotes the number of connected components.) $G$ is 1-supertough if, for all $S \subseteq V(G)$ with $|S| \geq 2$, $c(G - S) < |S|$.

The Delaunay tessellation, $DT(S)$, of a planar set of points $S$ is the unique graph with $V(G) = S$ such that the outer face is bounded by the convex hull of $S$, all vertices on the boundary of a common interior face are cocircular, the vertices of an interior face are exactly the points of $S$ lying on the circumcircle of the face, and no points of $S$ lie in the interior of a circumcircle of any interior face. $DT(S)$ is said to be nondegenerate if it is a triangulation and all convex