Computing the Geodesic Centers of a Polygonal Domain*

Sang Won Bae† Matias Korman‡,§ Yoshio Okamoto¶

Abstract

We present an algorithm that exactly computes the geodesic center of a given polygonal domain. The running time of our algorithm is $O(n^{12+\epsilon})$ for any $\epsilon > 0$, where n is the number of corners of the input polygonal domain. Prior to our work, only the very special case where a simple polygon is given as input has been intensively studied in the 1980s, and an $O(n \log n)$-time algorithm is known by Pollack et al. Our algorithm is the first one that handles general polygonal domains that may have one or more polygonal holes.

1 Introduction

A *polygonal domain* $\mathcal{P}$ with h holes and n corners is a connected and closed subset of $\mathbb{R}^2$ having h holes whose boundary consists of $h+1$ simple closed polygonal chains of n total line segments. The *diameter* and *radius* of $\mathcal{P}$, as a compact subset of $\mathbb{R}^2$, with respect to a certain metric d are the most natural and important measures of $\mathcal{P}$. The diameter with respect to d is defined to be the maximum distance over all pairs of points in $\mathcal{P}$, that is, $\max_{p,q \in \mathcal{P}} d(p, q)$, while the radius is defined to be the min-max value $\min_{p \in \mathcal{P}} \max_{q \in \mathcal{P}} d(p, q)$. A pair of points in $\mathcal{P}$ realizing the diameter is called a *diametral pair*, and a *center* is defined to be a point $c \in \mathcal{P}$ such that $\max_{q \in \mathcal{P}} d(c, q)$ is equal to the radius. In this paper we study one of the most natural metrics within a polygonal domain: the *geodesic distance* function $\mathrm{d}(p, q)$ for $p, q \in \mathcal{P}$ that measures the Euclidean length of a shortest path that connects p and q and stays within $\mathcal{P}$.

The problem of computing the diameter and center of a simple polygon (i.e., a polygonal domain with no holes) with respect to the geodesic distance d has been intensively studied in computational geometry since the early 1980s. Chazelle [3] gave the first algorithm for finding the geodesic diameter with running time $O(n^2)$, followed by an $O(n \log n)$-time algorithm by Suri [12]. Finally, Hershberger and Suri [6] presented a linear-time algorithm based on a fast matrix search technique. Asano and Toussaint [1] first addressed the problem of computing the Euclidean geodesic center of a simple polygon with an $O(n^4 \log n)$-time algorithm, and later Pollack, Sharir, and Rote [10] improved it to $O(n \log n)$ time. Since then, it has been a longstanding open problem whether the geodesic center can be computed in linear time, as also mentioned later by Mitchell [9].

On the other hand, the geodesic diameter of a domain having one or more holes is much less understood. What is known for general domains about this subject is a first algorithm by the authors that exactly computes the geodesic diameter of a polygonal domain in $O(n^{7.73})$ or $O(n^7(\log n + h))$ time [2]. In the preceding paper [2], we have shown that the diameter of a polygonal domain $\mathcal{P}$ may be determined by two points in the *interior* of $\mathcal{P}$, and this was one reason why the algorithm could not avoid such a high running time. Compare this with the fact that the geodesic diameter of a simple polygon is always determined by two corners, leading to much more efficient algorithms [6, 12].

One of the main differences between simple polygons and general domains lies on the difficulty to determine and discretize the search space. It is clear that in a simple polygon $\mathcal{P}$, each point that is farthest away from any fixed point $p \in \mathcal{P}$ must be a corner of $\mathcal{P}$ [1]. Thus, the diameter of any simple polygon should be realized by two corners of $\mathcal{P}$. This immediately gives $O(n^2)$ candidates that would determine the geodesic diameter. For computing the geodesic center, this helps a lot: even though the center may be an interior point of $\mathcal{P}$, it must be determined by two or three corners because only corners can achieve the maximum geodesic distance. The previous algorithms for simple polygons of course enjoy this nice behavior of the geodesic diameter and center. Unfortunately, this is not the case for general polygonal domains with one or more holes. This difference mainly causes the huge gap in the geodesic diameter algorithm between simple polygons [6] and general domains [2].

In this paper, we focus on computing the geodesic radius and center of a polygonal domain with holes, and present an algorithm that exactly computes it in

*Work by S.W. Bae was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2013R1A1A1A05006927). Work by Y. Okamoto was partially supported by Grant-in-Aid for Scientific Research from Ministry of Education, Science and Culture, Japan and Japan Society for the Promotion of Science, and the ELC project (Grant-in-Aid for Scientific Research on Innovative Areas, MEXT Japan).

†Dept. of Computer Science, Kyonggi University, Suwon, Korea. Email: `swbae@kgu.ac.kr`

‡National Institute of Informatics, Tokyo, Japan. Email: `korman@nii.ac.jp`

§JST, ERATO, Kawarabayashi Large Graph Project.

¶Dept. Communication Engineering and Informatics, University of Electro-Communications, Chofu, Tokyo, Japan Email: `okamotoy@uec.ac.jp`

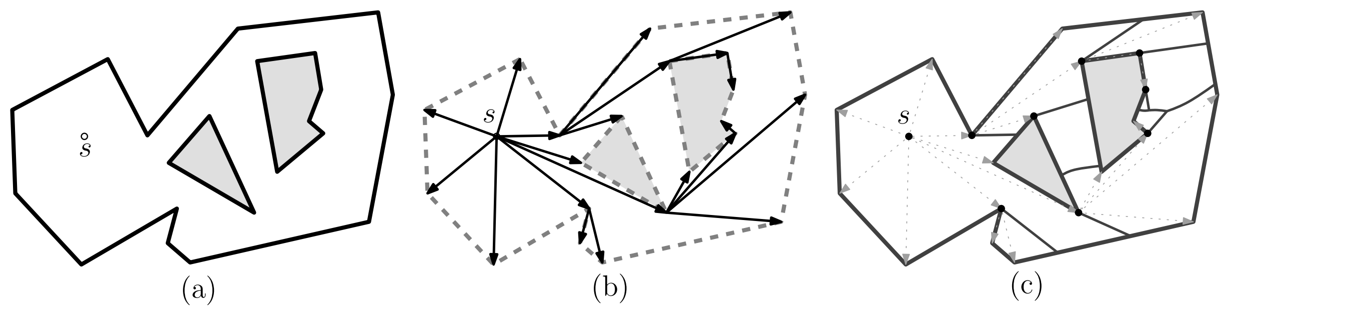

Figure 1: (a) Given a polygonal domain $\mathcal{P}$ with two holes and a source point $s \in \mathcal{P}$. (b) The shortest path tree $\mathsf{SPT}(s)$ on $V \cup \{s\}$ with root s whose edge are directed towards descendants. (c) The shortest path map $\mathsf{SPM}(s)$, whose edges are depicted by solid segments. Its edges, including the boundary of $\mathcal{P}$, are either straight or hyperbolic. Corners $v \in V$ with non-empty region $\sigma_s(v)$ are marked by black dots.

$O(n^{12+\epsilon})$ time. This is a first step from the result of Pollack et al. [10] towards general polygonal domains. The time complexity $O(n^{12+\epsilon})$ seems very high, but recall that the currently best algorithm that computes the geodesic diameter takes $O(n^{7.73})$ time [2]. Also, note that there was no known exact algorithm for general polygonal domains prior to our work, although the problem has been regarded to be natural and important [9, Open Problem 6].

The rest of the paper is organized as follows. After introducing preliminary definitions and concepts in Section 2, we list geometric observations that will be the base of our algorithms in Section 3. Section 4 is devoted to describe our algorithm. Finally, Section 5 concludes the paper with possible lines of research.

2 Preliminaries

Throughout the paper, we frequently use several topological concepts such as open and closed subsets, neighborhoods, and the boundary ∂A of a set A; unless stated otherwise, all of them are derived with respect to the standard topology on $\mathbb{R}^d$ with the Euclidean norm $\|\cdot\|$ for fixed $d \geq 1$. We also denote the straight line segment joining two points a, b by $\overline{ab}$.

A *polygonal domain* $\mathcal{P}$ with h holes and n corners[1] is a connected and closed subset of $\mathbb{R}^2$ with h holes whose boundary $\partial\mathcal{P}$ consists of $h+1$ simple closed polygonal chains of n total line segments. The boundary $\partial\mathcal{P}$ of a polygonal domain $\mathcal{P}$ is regarded as a series of *obstacles* so that any feasible path in $\mathcal{P}$ is not allowed to cross $\partial\mathcal{P}$. The *geodesic distance* $\mathrm{d}(p, q)$ between any two points p, q in a polygonal domain $\mathcal{P}$ is defined to be the Euclidean length of a shortest feasible path between p and q, where the *length* of a path is the sum of the Euclidean lengths of its segments. It is well known from earlier work [8] that there always exists a shortest feasible path between any two points $p, q \in \mathcal{P}$, and thus the geodesic distance function $\mathrm{d}(\cdot,\cdot)$ is well defined.

The *geodesic radius* $\mathrm{rad}(\mathcal{P})$ of $\mathcal{P}$ is defined to be the min-max quantity:

$$\mathrm{rad}(\mathcal{P}) = \min_{p\in\mathcal{P}} \max_{q\in\mathcal{P}} \mathrm{d}(p, q).$$

A *geodesic center* of $\mathcal{P}$ is a point $c \in \mathcal{P}$ such that

$$\max_{q\in\mathcal{P}} \mathrm{d}(c, q) = \mathrm{rad}(\mathcal{P}).$$

The purpose of this paper is to describe the first algorithm that computes the geodesic radius $\mathrm{rad}(\mathcal{P})$ together with the set of geodesic centers of a given polygonal domain $\mathcal{P}$.

2.1 Shortest path trees and shortest path maps

Let V be the set of all corners of $\mathcal{P}$ and π be a shortest path between any two points $s, t \in \mathcal{P}$. Then, it is easy to see that π is a polygonal chain that makes turns only at corners V of $\mathcal{P}$ [8]. That is, π is represented as a sequence of corners $\pi = (s, v_1, \ldots, v_k, t)$ for some $v_1, \ldots, v_k \in V$. Note that k may be zero; in this case, the shortest path π is the segment $\overline{st}$ connecting the two endpoints (and thus $\mathrm{d}(s,t) = \|s-t\|$). If two paths (with possibly different endpoints) induce the same sequence of corners $(v_1, \ldots, v_k)$, then they are said to have the same *combinatorial structure*.

Given a source point $s \in \mathcal{P}$, the *shortest path tree* $\mathsf{SPT}(s)$ of s is a tree spanning $V \cup \{s\}$ embedded inside $\mathcal{P}$ such that the unique path in $\mathsf{SPT}(s)$ from s to each corner of $\mathcal{P}$ is a shortest path in $\mathcal{P}$. See for example Figure 1(b), in which the edges of $\mathsf{SPT}(s)$ are directed towards descendants.

The *shortest path map* $\mathsf{SPM}(s)$ for a fixed $s \in \mathcal{P}$ is a decomposition of $\mathcal{P}$ into cells such that every point in a common cell can be reached from s by shortest paths of the same combinatorial structure. See Figure 1(c). Each cell $\sigma_s(v)$ of $\mathsf{SPM}(s)$ is associated with a corner

[1] We reserve the term “vertex” for a 0-dimensional face of subdivisions of a certain space.

$v \in V \cup \{s\}$ which is the last corner of $\pi(s,t)$ for any t in the cell $\sigma_s(v)$. We also define the cell $\sigma_s(s)$ as the set of points $t \in \mathcal{P}$ such that $\pi(s,t)$ passes through no corner of $\mathcal{P}$, so $\pi(s,t) = \overline{st}$. Each edge of $\mathsf{SPM}(s)$ either belongs to $\partial\mathcal{P}$ or is an arc on the boundary of two incident cells $\sigma_s(v_1)$ and $\sigma_s(v_2)$ determined by two corners $v_1, v_2 \in V \cup \{s\}$. Edges of the latter kind is a hyperbolic arc (if v_1 and v_2 are not adjacent in $\mathsf{SPT}(s)$). Moreover, there are two different shortest paths from s to any point on such edge (one via v_1 and the other via v_2). Finally, each vertex of $\mathsf{SPM}(s)$ is either a corner of $\mathcal{P}$, an endpoint of an edge of the second kind above, or a point $p \in \mathcal{P}$ incident to at least three faces $\sigma_s(v_1), \sigma_s(v_2), \sigma_s(v_3)$ for some corners $v_1, v_2, v_3 \in V \cup \{s\}$, yielding three different shortest paths from s.

The shortest path map $\mathsf{SPM}(s)$ has $O(n)$ cells, edges, and vertices in total, and can be computed in $O(n \log n)$ time using $O(n \log n)$ working space [7]. For more details on shortest path maps, see [7–9].

2.2 Path-length functions

For any point $p \in \mathcal{P}$, we define its *visibility region* as the set $\mathsf{VR}(p)$ of all points $q \in \mathcal{P}$ such that $\overline{pq} \subset \mathcal{P}$, that is, p and q *see* each other.

If $\pi(s,t) \neq \overline{st}$, then there are two corners $u, v \in V$ such that u and v are the first and last corners along $\pi(s,t)$ from s to t, respectively. Here, the path $\pi(s,t)$ is formed as the union of $\overline{su}$, $\overline{vt}$ and a shortest path $\pi(u,v)$ from u to v. Note that u and v are not necessarily distinct. In order to realize such a path, s must be visible from u and t visible from v. That is, $s \in \mathsf{VR}(u)$ and $t \in \mathsf{VR}(v)$,

We now define the *path-length function* $\mathrm{len}_{u,v}\colon \mathsf{VR}(u) \times \mathsf{VR}(v) \to \mathbb{R}$ for any fixed pair of corners $u, v \in V$ to be

$$\mathrm{len}_{u,v}(s,t) := \|s-u\| + \mathrm{d}(u,v) + \|v-t\|.$$

That is, $\mathrm{len}_{u,v}(s,t)$ represents the length of paths from s to t that have a common combinatorial structure; going straight from s to u, following a shortest path from u to v, and going straight to t. Also, unless $\mathrm{d}(s,t) = \|s-t\|$ (equivalently, $s \in \mathsf{VR}(t)$), the geodesic distance $\mathrm{d}(s,t)$ can be expressed as the pointwise minimum of some path-length functions:

$$\mathrm{d}(s,t) = \min_{u \in \mathsf{VR}(s)\cap V,\ v \in \mathsf{VR}(t)\cap V} \mathrm{len}_{u,v}(s,t).$$

By definition of shortest path map $\mathsf{SPM}(s)$ and its cells $\sigma_s(v)$, if $t \in \sigma_s(v)$ for some $v \in V$, then we have $\mathrm{d}(s,t) = \mathrm{len}_{u,v}(s,t)$, where $u \in V$ denotes the first corner along the shortest path from s to v, or equivalently, along the path from s to v in $\mathsf{SPT}(s)$.

3 Farthest Neighbors and Geodesic Centers

In this section we introduce several tools that will be useful for discretizing the search space for the centers of a polygonal domain.

For any point $p \in \mathcal{P}$, we let $\Phi(p)$ be the maximum geodesic distance we can obtain when we fix one point as p, that is,

$$\Phi(p) := \max_{q \in \mathcal{P}} \mathrm{d}(p,q).$$

We call a point $q \in \mathcal{P}$ a *farthest neighbor* of $p \in \mathcal{P}$ if $\mathrm{d}(p,q) = \Phi(p)$.

Observe that the geodesic radius of $\mathcal{P}$ is the minimum possible value of $\Phi(p)$ over all $p \in \mathcal{P}$, that is,

$$\mathrm{rad}(\mathcal{P}) = \min_{p \in \mathcal{P}} \Phi(p),$$

and a point that minimizes $\Phi(p)$ is a geodesic center of $\mathcal{P}$. Note that each geodesic center is determined by its farthest neighbors.

The following lemma suggests a principal rule to seek for farthest neighbors. Recall the definition of vertices of a shortest path map $\mathsf{SPM}(p)$.

Lemma 1 *For any point $p \in \mathcal{P}$, any farthest neighbor of p in $\mathcal{P}$ is a vertex of $\mathsf{SPM}(p)$.*

Proof. Suppose for contradiction that $q \in \mathcal{P}$ is a farthest neighbor of p, but q is not a vertex of $\mathsf{SPM}(p)$. Then, there exists a sufficiently short line segment L such that L is contained in the closure of some cell $\sigma_p(v)$ of $\mathsf{SPM}(p)$ for some $v \in V \cup \{p\}$ and contains q in its relative interior. This is always true even if q lies on an edge of $\mathsf{SPM}(p)$ since every edge of the shortest path map is either straight or hyperbolic. Then, the function $f(x) = \mathrm{d}(p,x)$ for $x \in L$ is represented as $f(x) = \mathrm{len}_{u,v}(p,x) = \|p-u\| + \mathrm{d}(u,v) + \|v-x\|$ for some $u \in V \cup \{p\}$. Observe that the function f is convex on L and has no plateau along its graph. Since q lies in the relative interior of L, there always exists a point $y \in L$ such that $f(y) > f(q)$, which contradicts the assumption that q is a farthest neighbor of p. □

This observation compares to the fact for simple polygons that any farthest neighbor of each point in a simple polygon P is a corner of P [10]. Vertices of shortest path maps $\mathsf{SPM}(p)$ may lie in the interior of $\mathcal{P}$. This means that a geodesic center may be determined by some interior points, whereas this never happens for simple polygons since farthest neighbors in any simple polygon must be its corners.

One can easily construct an instance of polygonal domain $\mathcal{P}$ such that the farthest neighbors of its geodesic center c indeed lie in the interior of $\mathcal{P}$. See Figure 2 for such an example.

The example presented in Figure 2 consists of three identical parts arranged in a symmetric way: each part

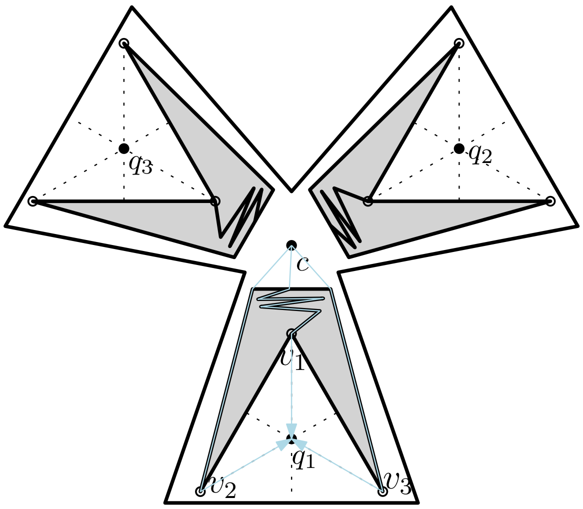

Figure 2: A polygonal domain instance with a unique geodesic center c. The three farthest neighbors q_1, q_2, and q_3 of the center c lie in the interior of the domain. Observe that there are three distinct shortest paths between the center c and each of its farthest neighbors, and thus 9 shortest paths of equal length in total.

contains two holes that almost fit together forming a very narrow corridor between them. We claim that c is the unique geodesic center determined by three interior points q_1, q_2, and q_3. Note that q_1, q_2, and q_3 are the three farthest neighbors of c. Moreover, each q_i is a vertex of the shortest path map $\mathsf{SPM}(c)$ for c. That is, there exist three combinatorially distinct shortest paths from c to q_i. (The paths from c to q_1 are depicted in Figure 2.) Moreover, no other vertex of $\mathsf{SPM}(c)$ is farther from c than q_i. The point c is thus a geodesic center of this polygonal domain. Also, due to symmetry in the construction, we observe that no point in $\mathcal{P}$ other than c can be closer to all of q_1, q_2, and q_3 at the same time. This implies that c is the unique geodesic center.

Note that the example is symmetric and seems degenerate in a sense, but such a degeneracy can be removed easily by means of perturbation. This is possible because the center c is "stable" in the sense that if we perform a sufficiently small perturbation on the corners of the domain, its geodesic center and the corresponding farthest neighbors will not change too much.

4 Algorithm

In this section, we describe our algorithm for computing the geodesic radius $\mathrm{rad}(\mathcal{P})$ and the geodesic centers of a given polygonal domain $\mathcal{P}$. Recall that the problem of computing the geodesic radius and center can be seen as a minimization problem with objective function Φ over $\mathcal{P}$. Thus, our approach is to decompose $\mathcal{P}$ into cells, and find one (or more) candidate centers in each cell.

For any subset $\sigma \subseteq \mathcal{P}$ of the domain $\mathcal{P}$, we call the minimum value of $\Phi(p)$ over $p \in \sigma$ the *σ-constrained geodesic radius*, and each point in σ that attains the minimum is a *σ-constrained geodesic center*. Clearly, in any decomposition $\{\sigma_1, \sigma_2, \ldots\}$ of $\mathcal{P}$, the geodesic radius is the minimum of σ_i-constrained geodesic radii over all i, and the points that attain the minimum value are the geodesic centers of $\mathcal{P}$.

In this paper, we adopt the *SPM-equivalence* decomposition. The SPM-equivalence decomposition of a polygonal domain has been first described by Chiang and Mitchell [4] to devise efficient data structures that support two-point queries for Euclidean shortest paths.

The SPM-equivalence decomposition $\mathcal{A}_{\mathsf{SPM}}$ of a polygonal domain $\mathcal{P}$ subdivides $\mathcal{P}$ into cells such that for all points s in a common cell σ of $\mathcal{A}_{\mathsf{SPM}}$, their shortest path maps $\mathsf{SPM}(s)$ are topologically equivalent. More precisely, two shortest path maps $\mathsf{SPM}(s_1)$ and $\mathsf{SPM}(s_2)$ are said to be *topologically equivalent* if their underlying plane graphs are isomorphic. Chiang and Mitchell [4] show that the decomposition $\mathcal{A}_{\mathsf{SPM}}$ has $O(n^{10})$ complexity and can be computed in $O(n^{10} \log n)$ time.

An additional property of this subdivision is that, for a fixed cell σ of $\mathcal{A}_{\mathsf{SPM}}$, every element of $\mathsf{SPM}(s)$ (such as vertices and edges) can be explicitly stored in the form of algebraic functions of $s \in \sigma$. In this manner, the shortest path map $\mathsf{SPM}(s)$ for all $s \in \sigma$ is parameterized, as Chiang and Mitchel [4] also discussed.

Now, pick any cell σ of $\mathcal{A}_{\mathsf{SPM}}$. We are in particular interested in the vertices $v_1, v_2, \ldots, v_m$ of $\mathsf{SPM}(s)$, as functions of s from σ to a point $v_i(s) \in \mathbb{R}^2$, where m is the number of vertices of $\mathsf{SPM}(s)$. Recall that the vertices v_i of $\mathsf{SPM}(s)$ include the corners V of $\mathcal{P}$; If for some $i \leq m$ we have $v_i \in V$, then $v_i(s)$ will be a constant function that always maps to a unique corner of $\mathcal{P}$.

For any $i \in \{1, 2, \ldots, m\}$, we define the function $f_i : \sigma \to \mathbb{R}$ to be $f_i(s) = \mathrm{d}(s, v_i(s))$ for $s \in \sigma$. This function maps s to the geodesic distance from s to $v_i(s)$. We then consider the upper envelope $\max_i f_i(s)$ of the m functions, which maps s to its maximum geodesic distance over all the vertices $v_i(s)$ of $\mathsf{SPM}(s)$. Lemma 1 guarantees that we can find a farthest neighbor of s among the $v_i(s)$, and thus it holds that

$$\Phi(s) = \max_{i=1,2,\ldots,m} f_i(s).$$

That is: in order to find the σ-constrained geodesic radius and center, it suffices to compute the upper envelope of the m functions f_i.

For this purpose, we consider the shortest paths from $s \in \sigma$ to each $v_i(s)$ and then obtain the description of the functions f_i. There are two different cases: either $v_i(s)$ is always *visible* from s or $v_i(s)$ is always *invisible* from s. Since σ is a cell of the SPM-equivalence decomposition $\mathcal{A}_{\mathsf{SPM}}$, there is no other situation such that $v_i(s_1)$ is visible while $v_i(s_2)$ is invisible for $s_1, s_2 \in \sigma$; otherwise,

Algorithm GeodesicCenter($\mathcal{P}$)
  Compute the SPM-equivalence decomposition $\mathcal{A}_{\mathsf{SPM}}$ of $\mathcal{P}$.
  for each cell σ of $\mathcal{A}_{\mathsf{SPM}}$ **do**
    Specify the combinatorial structure of the shortest path maps $\mathsf{SPM}(s)$ for $s \in \sigma$.
    Identify the parameterized equations of the vertices of $\mathsf{SPM}(s)$.
    Let $v_1(s), \ldots, v_m(s)$ be the parameterized points identified by the above step.
    Let $f_i(s) := \mathrm{d}(s, v_i(s))$ be the m bivariate functions for $s \in \sigma$.
    Compute the upper envelope $\mathcal{U}_\sigma$ of the m graphs $\{z = f_i(s)\}$.
    Find the points c_σ with the lowest z-coordinate in $\mathcal{U}_\sigma$.
    Store c_σ as the σ-constrained geodesic centers with its z-value.
  end for
  return the constrained geodesic centers c_σ having smallest z-value.
  return the z-value as the radius of $\mathcal{P}$
end Algorithm

Figure 3: An $O(n^{12+\epsilon})$-time algorithm for computing the geodesic center of a polygonal domain

it causes a change in their shortest path trees and thus a change of their shortest path maps. Thus, we shall call each v_i *visible* or *invisible* accordingly. By the same reason, for any corner $v \in V$ of $\mathcal{P}$, v is always visible or always invisible from s over all $s \in \sigma$. By an abuse of notation, we write $\mathsf{VR}(\sigma)$ to denote the set of corners that are visible from $s \in \sigma$.

Lemma 2 *Let σ and the v_i's be defined as above. The vertex v_i is visible if and only if v_i maps $s \in \sigma$ to a corner $v \in V$ of $\mathcal{P}$ such that $v \in \mathsf{VR}(\sigma)$.*

Proof. Assume that v_i is visible. That is, $v_i(s)$ and s always see each other for any $s \in \sigma$, so the shortest path from s to $v_i(s)$ is just the straight line segment $\overline{sv_i(s)}$. We claim that $v_i(s) = v$ for some corner $v \in V$ of $\mathcal{P}$. As discussed in Section 2.1, each vertex of the shortest path map $\mathsf{SPM}(s)$ falls into one of three cases. Among these three cases, $v_i(s)$ must fall into the first one: $v_i(s)$ is a corner $v \in V$ of $\mathcal{P}$ for all $s \in \sigma$ since $v_i(s)$ admits only a unique shortest path from s, $\overline{sv_i(s)}$. Therefore, we have $v \in \mathsf{VR}(\sigma)$.

The other direction is easy to see. If $v_i(s) = v$ for some corner $v \in V$ with $v \in \mathsf{VR}(\sigma)$, then $v_i(s)$ is always visible from any $s \in \sigma$. Thus, the lemma follows. ◻

Lemma 3 *Let σ and the v_i's be declared as above. For each $i = 1, \ldots, m$, it holds that*

$$f_i(s) = \begin{cases} \|s - t\| & \text{if } v_i \text{ is visible,} \\ \mathrm{len}_{u_i, w_i}(s, v_i(s)) & \text{otherwise,} \end{cases}$$

where $u_i, w_i \in V$ are corners of $\mathcal{P}$ uniquely determined by v_i.

Proof. We consider the two cases separately. Suppose first that v_i is visible. Then, by Lemma 2, $v_i(s) = v$ for some corner $v \in V$ of $\mathcal{P}$ such that $v \in \mathsf{VR}(\sigma)$. This directly implies the lemma.

Next, assume the latter case where v_i is invisible. Pick any point $s_0 \in \sigma$, and consider a shortest path π from s_0 to $v_i(s_0)$. Let $u_i \in V$ and $w_i \in V$ be the first and the last corners of $\mathcal{P}$ along π. Since v_i is invisible, any shortest path from s_0 to $v_i(s_0)$ cannot be the straight line segment $\overline{s_0 v_i(s_0)}$ by Lemma 2, so such corners u_i and w_i must exist. This implies that $f_i(s_0) = \mathrm{d}(s_0, v_i(s_0)) = \mathrm{len}_{u_i, w_i}(s_0, v_i(s_0))$. By definition of the SPM-equivalence decomposition $\mathcal{A}_{\mathsf{SPM}}$, for any $s \in \sigma$, there exists a shortest path from s to $v_i(s)$ whose first corner is u_i and last corner is w_i. Hence, the lemma follows. ◻

By combining the above observations, we are now able to describe our algorithm that exactly computes the geodesic radius and centers of a given polynomial domain. See Figure 3. We finally conclude with our main theorem.

Theorem 4 *The algorithm described in Figure 3 correctly computes the geodesic radius and centers of a polygonal domain with n corners in $O(n^{12+\epsilon})$ time for any positive ϵ.*

Proof. The correctness follows from the discussion above. In order to show the time bound, we need an efficient tool to compute the upper envelope of functions. Given a collection of N algebraic surface patches in $\mathbb{R}^d$, we can compute their lower (or upper) envelope in $O(N^{d-1+\epsilon})$ time using the algorithms of Halperin and Sharir [5] (for $d = 3$) or Sharir [11] (for $d > 3$). Note that the complexity of the resulting envelope is bounded by $O(N^{d-1+\epsilon})$.

Recall that the coordinates of each vertex $v_i(s)$ of $\mathsf{SPM}(s)$ is an algebraic function [4]. Lemma 3 implies the functions f_i are algebraic, too. Thus, we can apply the above algorithms to compute the upper envelope $\mathcal{U}_\sigma$ of the graphs of f_i, which is equivalent to the function Φ.

In our case, we have $N = O(n)$, since any shortest path map $\mathsf{SPM}(s)$ has $O(n)$ complexity. Each function f_i has two arguments (i.e., the coordinates of s within σ), so the graph of f_i lies in three-dimensional space. Thus, the upper envelope $\mathcal{U}_\sigma$ of the functions f_i can be computed in $O(n^{2+\epsilon})$ for any positive ϵ. Once the envelope is computed, we can find the points with the lowest z-coordinate in $\mathcal{U}_\sigma$ in the same time bound by traversing all faces of the envelope $\mathcal{U}_\sigma$. The point that minimizes $\mathcal{U}_\sigma$ is a candidate for center, and its image will be its corresponding radius.

Thus, we spend $O(n^{2+\epsilon})$ time per cell σ of $\mathcal{A}_{\mathsf{SPM}}$. Since $\mathcal{A}_{\mathsf{SPM}}$ consists of $O(n^{10})$ cells, we obtain the claimed time bound $O(n^{12+\epsilon})$. □

5 Concluding Remarks

We have presented the first algorithm that exactly computes the geodesic radius and centers of a general polygonal domain with holes. The time complexity of our algorithm is quite big, but still remarkable as the first nontrivial upper bound. Note that this high complexity $O(n^{12+\epsilon})$ heavily relies on the complexity of the SPM-equivalence decomposition $\mathcal{A}_{\mathsf{SPM}}$. The currently best upper bound of the complexity of $\mathcal{A}_{\mathsf{SPM}}$ is $O(n^{10})$, and it is known how to construct a polygonal domain whose decomposition $\mathcal{A}_{\mathsf{SPM}}$ has $\Omega(n^4)$ complexity [4].

Besides that, the algorithm could be improved by exploiting a coarser subdivision, such as the *SPT-equivalence decomposition* [4]. The SPT-equivalence decomposition $\mathcal{A}_{\mathsf{SPT}}$ is obtained by overlaying the n shortest path maps $\mathsf{SPM}(v)$ for corners $v \in V$ of $\mathcal{P}$. Then, it is guaranteed that the shortest path trees $\mathsf{SPT}(s)$ for all s in each cell of $\mathcal{A}_{\mathsf{SPT}}$ are isomorphic. The complexity of $\mathcal{A}_{\mathsf{SPT}}$ is $O(n^4)$, and thus exploiting $\mathcal{A}_{\mathsf{SPT}}$ instead of $\mathcal{A}_{\mathsf{SPM}}$ would be helpful. Ideally, we would want an algorithm that can compute the σ-constrained geodesic radius for a cell σ of $\mathcal{A}_{\mathsf{SPT}}$ in $o(n^{8+\varepsilon})$ (so that the running time improves Theorem 4). However, all of our attempts to do so need significantly more than $\Omega(n^8)$, which lead to slower algorithms.

References

[1] T. Asano and G. Toussaint. Computing the geodesic center of a simple polygon. Technical Report SOCS-85.32, McGill University, 1985.

[2] S. W. Bae, M. Korman, and Y. Okamoto. The geodesic diameter of polygonal domains. *Discrete Comput. Geom.*, 50(2):306–329, 2013.

[3] B. Chazelle. A theorem on polygon cutting with applications. In *Proc. 23rd Annu. Sympos. Found. Comput. Sci. (FOCS)*, pages 339–349, 1982.

[4] Y.-J. Chiang and J. S. B. Mitchell. Two-point Euclidean shortest path queries in the plane. In *Proc. 10th ACM-SIAM Sympos. Discrete Algorithms (SODA)*, pages 215–224, 1999.

[5] D. Halperin and M. Sharir. New bounds for lower envelopes in three dimensions, with applications to visbility in terrains. *Discrete & Computational Geometry*, 12:313–326, 1994.

[6] J. Hershberger and S. Suri. Matrix searching with the shortest path metric. *SIAM J. Comput.*, 26(6):1612–1634, 1997.

[7] J. Hershberger and S. Suri. An optimal algorithm for Euclidean shortest paths in the plane. *SIAM J. Comput.*, 28(6):2215–2256, 1999.

[8] J. S. B. Mitchell. Shortest paths among obstacles in the plane. *Internat. J. Comput. Geom. Appl.*, 6(3):309–331, 1996.

[9] J. S. B. Mitchell. Geometric shortest paths and network optimization. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 633–701. Elsevier, 2000.

[10] R. Pollack, M. Sharir, and G. Rote. Computing the geodesic center of a simple polygon. *Discrete Comput. Geom.*, 4(6):611–626, 1989.

[11] M. Sharir. Almost tight upper bounds for lower envelopes in higher dimensions. *Discrete Comput. Geom.*, 12:327–345, 1994.

[12] S. Suri. Computing geodesic furthest neighbors in simple polygons. *J. Comput. Syst. Sci.*, 39(2):220–235, 1989.