Computing the Optimal Bridge between Two Polygons

Sung Kwon Kim
Dept. of Computer Science and
Engineering, Chung-Ang Univ., Korea
skkim@cau.ac.kr

Chan-Su Shin
Dept. of Computer Science
KAIST, Taejon, Korea
cssin@jupiter.kaist.ac.kr

June 21, 2000

Abstract

Let $\mathcal{P}$ and $\mathcal{Q}$ be disjoint polygons in the plane. We consider problems of finding an optimal bridge (p, q), $p \in \partial\mathcal{P}$ and $q \in \partial\mathcal{Q}$, such that the length of the longest path from a point in $\mathcal{P}$, passing through the bridge (p, q), to a point $\mathcal{Q}$ is minimized. We propose efficient algorithms for three problems according to whether $\mathcal{P}$ and $\mathcal{Q}$ are convex or not. The algorithms developed can be easily extended to more general problems in which a set S of points in $\mathcal{P}$ and a set T of points in $\mathcal{Q}$ are given, and the goal is to find a bridge (p, q) such that minimizes the length of the longest path connecting a point in S, passing through (p, q), to a point in T. These problems are motivated from the bridge construction between two islands (or the canal construction between two lakes).

Keywords: Computational geometry, min-max optimization, polygons.

1 Introduction

Let $\mathcal{P}$ be a simple polygon in the plane. We shall refer to $\mathcal{P}$ as the union of its interior and boundary. A *geodesic shortest path* $\mathrm{SP}(p, q; \mathcal{P})$ from a point $p \in \mathcal{P}$ to a point $q \in \mathcal{P}$ is a shortest path connecting p and q avoiding the exterior of $\mathcal{P}$. We denote the length of $\mathrm{SP}(p, q; \mathcal{P})$, i.e., the geodesic distance between p and q in $\mathcal{P}$, by $|\mathrm{SP}(p, q; \mathcal{P})|$. To denote the Euclidean distance between p and q, we use $d(p, q)$. If $\mathcal{P}$ is convex, then $\mathrm{SP}(p, q; \mathcal{P})$ is a line segment from p to q, and $|\mathrm{SP}(p, q; \mathcal{P})|$ is identical to $d(p, q)$; however, if $\mathcal{P}$ is non-convex, it is a path from p to q consisting of several line segments, and $|\mathrm{SP}(p, q; \mathcal{P})|$ is the sum of lengths of the segments. Throughout the paper, we will use $\mathrm{SP}(p, q)$ simply, instead of $\mathrm{SP}(p, q; \mathcal{P})$, if understood.

Let $\mathcal{P}$ and $\mathcal{Q}$ be two disjoint polygons of n and m vertices with boundaries $\partial\mathcal{P}$ and $\partial\mathcal{Q}$, respectively. We now want to construct a *bridge* (p, q), a line segment connecting $\mathcal{P}$ and $\mathcal{Q}$, where $p \in \partial\mathcal{P}$ and $q \in \partial\mathcal{Q}$ are visible each other, that minimizes

$$\max_{p' \in \partial\mathcal{P}}(|\mathrm{SP}(p', p)|) + d(p, q) + \max_{q' \in \partial\mathcal{Q}}(|\mathrm{SP}(q, q')|).$$

If we denote the set of vertices of $\mathcal{P}$ and $\mathcal{Q}$ by $V(\mathcal{P})$ and $V(\mathcal{Q})$, respectively, then, by a simple observation, the above function is equivalent to the following one:

$$\max_{v \in V(\mathcal{P})}(|\mathrm{SP}(v,p)|) + d(p,q) + \max_{w \in V(\mathcal{Q})}(|\mathrm{SP}(q,w)|).$$

If $f_{\mathcal{P}}(p)$ denotes the (geodesic) furthest vertex of $V(\mathcal{P})$ from a point $p \in \mathcal{P}$, then the first and third terms can be simplified:

$$|\mathrm{SP}(f_{\mathcal{P}}(p),p)| + d(p,q) + |\mathrm{SP}(q, f_{\mathcal{Q}}(q))|.$$

We shall refer to a bridge (p,q) that minimizes the function as an *optimal bridge* (p,q) *between* $\mathcal{P}$ and $\mathcal{Q}$, and the path from $f_{\mathcal{P}}(p)$ of $\mathcal{P}$, passing through the optimal bridge (p,q), to $f_{\mathcal{Q}}(q)$ of $\mathcal{Q}$ as an *optimal path* of $\mathcal{P}$ and $\mathcal{Q}$.

Kim [8] presented algorithms for computing an optimal bridge when its end-points are *restricted* to the vertices of $\mathcal{P}$ and $\mathcal{Q}$. He considered three problems according to whether $\mathcal{P}$ and $\mathcal{Q}$ are convex or not. Recently, Cai et al. [5] proposed an $O((n+m)^2 \log(n+m))$ time algorithm for computing an optimal bridge between two convex polygons, and also presented a 2-factor approximation algorithm running in linear time. As a simplified version of the problem, Kim et al. [7] considered a problem of computing an optimal bridge crossing a river with two parallel line boundaries so that the length of the longest path connecting two towns (i.e., points) on opposite sides of the river is minimized. They proposed a linear-time algorithm. To our best knowledge, no other algorithms have been known. In this paper, we consider the three bridge problems according to whether polygons are convex or not, and our results are summarized in Table 1.

Problems	$\mathcal{P}$	$\mathcal{Q}$	Previous results	Our results
convex-convex	convex	convex	$O((n+m)^2 \log(n+m))$ [5]	$O(n+m)$
simple-convex	simple	convex	$-$	$O((n+m)\log(n+m))$
simple-simple	simple	simple	$-$	$O(nm + m\log m)$ for $m \geq n$

Table 1: The summary of the results.

We can consider a more general problem as follows: Given a set S of points in $\mathcal{P}$ and a set T of points in $\mathcal{Q}$, find an optimal bridge (p,q) for $p \in \partial\mathcal{P}$ and $q \in \partial\mathcal{Q}$ that minimizes

$$\max_{s \in S}(|\mathrm{SP}(s,p)|) + d(p,q) + \max_{t \in T}(|\mathrm{SP}(q,t)|).$$

Note that if S (resp., T) is $V(\mathcal{P})$ (resp., $V(\mathcal{Q})$), then the problem becomes identical to our problem. Actually, we can directly apply our algorithms in Table 1 to solve the general problems with the same time bounds, only increasing a $\log n$-factor when two polygons are convex. It is worthwhile to note that the 2-approximation algorithm [5] presented for the convex-convex problem still holds for the general problems where at least one of the polygons is simple.

Before explaining our algorithms in detail, we need to mention the furthest Voronoi diagram. A *furthest Voronoi diagram* for a set S of points in the plane partitions the whole plane into unbounded convex regions, called *furthest Voronoi regions*, such that every point in a furthest Voronoi region has the same furthest point in S. A point in S is associated with a furthest Voronoi region. Note that the points on the convex hull of S only contribute to the diagram, thus if a point is not extreme, its associated Voronoi region is empty. It is well known that the furthest Voronoi diagram can be computed in $O(|S|)$ time when the convex hull of S is already computed [1].

If the points of S are in a simple polygon $\mathcal{P}$ (possibly, on $\partial\mathcal{P}$), and if the length of the geodesic shortest path in $\mathcal{P}$ is used for measuring the distance between two points in S, then the furthest Voronoi diagram for S in $\mathcal{P}$ is defined; for more precise definitions, refer to [2]. The following lemma is important in our algorithms.

Lemma 1 [2] *Given a set S of k points in a simple n-sided polygon $\mathcal{P}$, the following properties for the furthest Voronoi diagram for S in $\mathcal{P}$ hold:* (i) *The intersection of a non-empty Voronoi region with $\partial\mathcal{P}$ is connected,* (ii) *the diagram has $O(n+k)$ Voronoi vertices, edges, and regions in $\mathcal{P}$, and* (iii) *the diagram is computed in $O((n+k)\log(n+k))$ time and $O(n+k)$ space.*

2 Convex-convex problem

Let $\mathcal{P}$ and $\mathcal{Q}$ be two convex polygons of n and m vertices, respectively. Assume, without loss of generality, that there is a vertical line separating $\mathcal{P}$ and $\mathcal{Q}$, and $\mathcal{P}$ is to the left of $\mathcal{Q}$. Let ℓ and ℓ' be two supporting lines for $\mathcal{P}$ and $\mathcal{Q}$ such that both $\mathcal{P}$ and $\mathcal{Q}$ entirely lie in the lower side of ℓ and in the upper side of ℓ'. Then $\partial\mathcal{P}$ is divided into two (left and right) chains by two intersection points $\ell\cap\partial\mathcal{P}$ and $\ell'\cap\partial\mathcal{P}$; $\partial\mathcal{Q}$ is similarly divided. Two end-points of a bridge should be visible each other, thus we are interested only in the points on the right chain R of $\partial\mathcal{P}$ and the points on the left chain L of $\partial\mathcal{Q}$. Let $V(R)$ and $V(L)$ denote the sets of vertices of R and L, including their end-points, respectively. In the remainder of the section, we assume that the vertices in $V(R)$ and $V(L)$ are ordered from top to bottom.

2.1 Partitioning R and L

The preliminary step is to compute two furthest Voronoi diagrams for the vertices of $\mathcal{P}$ and for those of $\mathcal{Q}$. We will denote them by $\mathrm{FVD}(\mathcal{P})$ and $\mathrm{FVD}(\mathcal{Q})$, respectively. The diagrams can be constructed in linear time [1] as both polygons are convex.

Consider a point p on R. If we build a bridge with p as its one end-point, then the other end-point on L must be visible from p. However, all points on L are always not visible from p. Thus it is convenient to partition R into line segments R_i so that all points on R_i have (i) the same furthest vertex $f_{\mathcal{P}}(R_i)$ [1] and (ii) the same visible portion W_i of L.

To satisfy the condition (i), we compute $R\cap\mathrm{FVD}(\mathcal{P})$, the intersection of R with the Voronoi edges of $\mathrm{FVD}(\mathcal{P})$. Denote the collection of such (*red*) intersections by I_r. Clearly, the points between two consecutive red intersections have the same furthest vertex. By property (i) of Lemma 1, $|I_r| = O(n)$. Thus the points in $I_r \cup V(R)$ partition R into line segments satisfying the condition (i).

For the condition (ii), we use the well-known partitioning method in geometric algorithms (e.g., see [4]). For completeness of the description, we will briefly explain it.

Assume that the vertices $u_0, u_1, \ldots, u_k$ of $V(L)$ are ordered from top to bottom. For every edge $e_i = (u_i, u_{i+1})$ of L for $0 \le i \le k-1$, define two rays γ_i and $\bar{\gamma}_i$ extending e_i where γ_i emanates

[1] We will use $f_{\mathcal{P}}(C)$ to denote the geodesic furthest vertex of points on a chain $C \in \partial\mathcal{P}$ only when every point on C has the same furthest vertex.

from u_i to u_{i+1} and $\bar{\gamma}_i$ emanates from u_{i+1} to u_i. Compute a *blue* intersection of R with γ_i (resp., $\bar{\gamma}_i$) for each i, and denote the ordered set of the intersections by I_γ (resp., $I_{\bar{\gamma}}$). Some rays might have no intersections. I_γ and $I_{\bar{\gamma}}$ can be computed in $O(|R| + |L|)$ time by scanning R and L twice; see [4] for details. Now merge I_γ and $I_{\bar{\gamma}}$ from top to bottom, and denote the merged set by I_b. Finally, number the points in the ordered set $I = I_r \cup I_b \cup V(R)$ as $p_1, p_2, \cdots, p_{n'}$ from top to bottom.

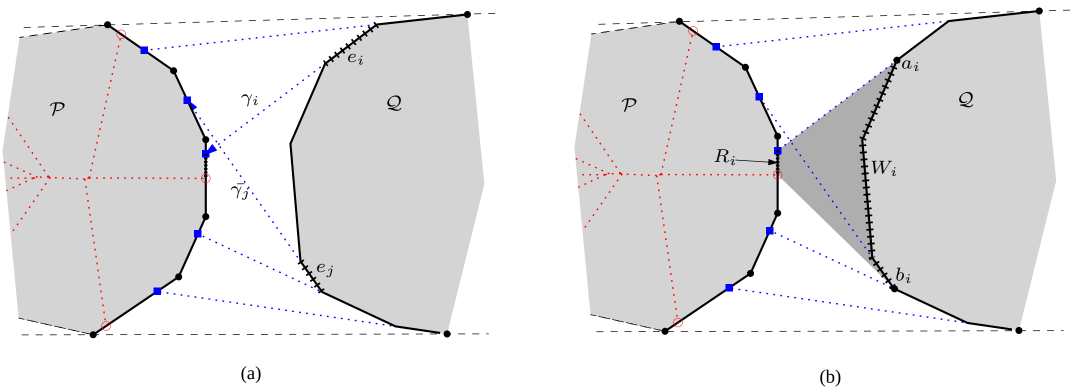

Figure 1: (a) Red points (marked as crosses), blue points (marked as squares), and vertices (marked as black circles) on R. (b) Definitions of a_i, b_i, W_i for R_i.

We partition R into line segments $R_i = (p_i, p_{i+1})$ for $1 \le i < n'$. Since $|I| = O(n+m)$, the number of the resulting segments is $O(n + m)$. This partition satisfies the condition (ii); otherwise, i.e., if points on R_i have different visible portions, there should be a point $q \in L$ which $p \in R_i$ can see, but $p' \in R_i$ cannot. This implies the intersection of R with the ray extending the edge of L containing q must lie between p and p' (excluding p and p'). Thus R_i would be split into smaller segments.

Let us now explain how to compute the visible portion W_i for each R_i in $O(n + m)$ total time. By the convexity of R and L, W_i is connected. A blue point p of $I_b = I_\gamma \cup I_{\bar{\gamma}}$ is associated with an edge of L whose extended line meets R at p. We call the end-vertex of the edge nearer to p a *border* of p. Let a_i be the border of the bottommost point of I_γ above or on p_i and let b_i be the border of the topmost point of $I_{\bar{\gamma}}$ below or on p_i. Since every point either above a_i or below b_i on L is not visible from any point on R_i, W_i is defined as a chain on L between a_i and b_i. If a line containing R_i intersects L at a point c_i which is between a_i and b_i, then either of a_i and b_i, say b_i, must not be visible from any point on R_i. This means the points below c_i are not visible from R_i. Accordingly, W_i is defined to be a portion between a_i and c_i. If the line does not intersect, then W_i becomes a portion between a_i and b_i.

To determine the end-points of W_i, we have to know three points a_i, b_i, and c_i. The first two points can be easily obtained as a by-product during the computation of I_b. The third one can be obtained in $O(|R| + |L|)$ time for each R_i by extending each edge of R until it intersects L like the computation of I_b. As a result, we can partition R into $O(n + m)$ R_i's, and compute W_i for each R_i in $O(|R| + |L|) = O(n + m)$ time.

Similarly, we define the partition of L into line segments $L_j = (q_j, q_{j+1})$, where $q_1, \ldots, q_{m'}$ are the ordered sequence of the red, blue intersections, and the vertices of L. Then the end-points of W_i are now represented as two points in $\{q_j\}$. In what follows, we assume that W_i is a portion of

L between $q_{\alpha(i)}$ and $q_{\beta(i)}$, where $q_{\alpha(i)}$ (resp., $q_{\beta(i)}$) is the topmost (resp., bottommost) visible point from every point on R_i. In other words, the consecutive segments $L_{\alpha(i)}, \ldots, L_{\beta(i)-1}$ are visible from all points on R_i. Clearly, $\alpha(i) < \beta(i)$.

2.2 Finding the optimal bridge

We imagine that there is an $(n'-1) \times (m'-1)$ matrix M whose rows represent R_i's and columns represent L_j's. Remember that $n' = O(n+m)$ and $m' = O(n+m)$. Each entry $M(i,j)$ is defined as the length of the optimal path through R_i and L_j. The optimal path is defined only when L_j is contained in W_i, i.e., when $\alpha(i) \leq j < \beta(i)$. Thus, the entries $M(i,j)$ for $1 \leq j < \alpha(i)$ and $\beta(i) \leq j < m'$ are undefined. Furthermore, since $\alpha(i) \leq \alpha(i')$ and $\beta(i) \leq \beta(i')$ for any $i < i'$, the defined entries of any row and of any column appear consecutively. Note that each entry $M(i,j)$ can be computed in constant time because there are only nine possible configurations for the optimal path as in Figure 2.

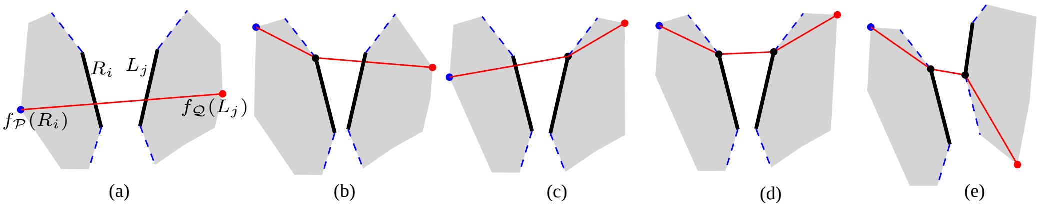

Figure 2: Only five ones among nine configurations for an optimal bridge through R_i and L_j are illustrated. Symmetric ones are omitted.

For a fixed row i, consider an ordered sequence S_i of values, $M(i,\alpha(i)), M(i,\alpha(i)+1), \ldots, M(i,\beta(i)-1)$. The sequence S_i is *unimodal*, meaning that S_i has at most one local minima. (Such unimodality will be proven in Lemma 2 below.) Let $\gamma(i)$ be the column index at which the minimum of S_i occurs. Then we have that for any two rows $i < j$, $\gamma(i) \leq \gamma(j)$. To prove this by contradiction, assume that $\gamma(i) > \gamma(j)$. Then the optimal bridge through R_i and $L_{\gamma(i)}$ must intersect the optimal bridge through R_j and $L_{\gamma(j)}$. The triangle inequality implies that $M(i,\gamma(j)) < M(i,\gamma(i))$ and $M(j,\gamma(i)) < M(j,\gamma(j))$, which are contradictions.

As a result, each sequence S_i is unimodal and $\gamma(i)$ is increasing as i increases. From these facts, we can find a minimum of each row by a simple linear scan as follows: (i) Evaluate the entries of the first row, starting from $M(1,\alpha(1)), \ldots, M(1,\beta(1)-1)$ one by one. This evaluation stops when the minimum index $\gamma(1)$ is reached. (Since S_1 is unimodal, $\gamma(1)$ can be easily identified by comparing the neighboring entries of $M(1,\gamma(1))$.) (ii) Since $\gamma(i)$ for $i > 1$ is no larger than $\gamma(i-1)$, we continue to find $\gamma(i)$ by evaluating entries at row i, starting from $M(i,\gamma(i-1))$. Eventually all $\gamma(i)$ are computed by a "staircase" scan in M, which takes $O(m+n)$ time.

What remains is to prove S_i for a fixed R_i is unimodal. For it, we need several notations. For a fixed $x \in W_i$, let $\Pi(x)$ be an optimal path through R_i and x, and let $\ell(x)$ be the length of $\Pi(x)$. Let $p(x) = R_i \cap \Pi(x)$. Then $\ell(x) = d(f_{\mathcal{P}}(R_i), p(x)) + d(p(x), x) + d(x, f_{\mathcal{Q}}(x))$. We will prove that $\ell(x)$ is unimodal over W_i, which immediately implies that S_i is unimodal.

Lemma 2 *$\ell(x)$ over $x \in W_i$ is unimodal.*

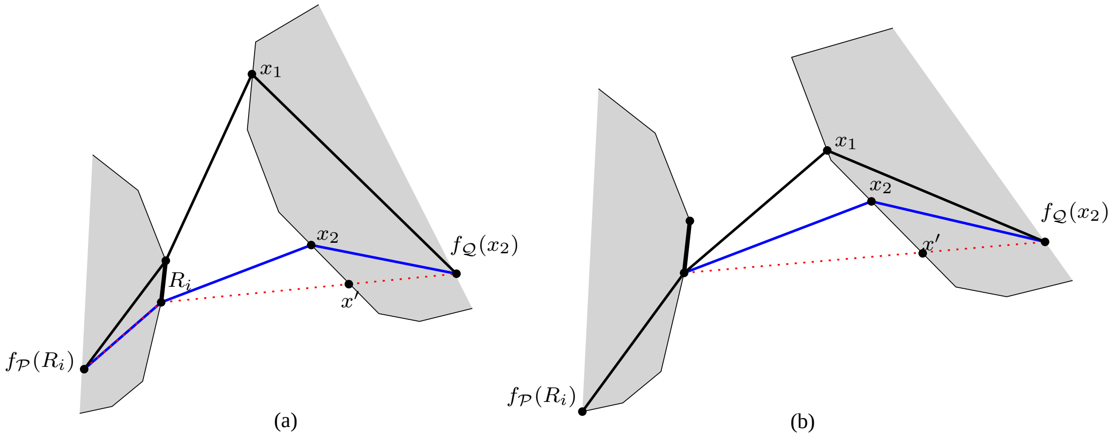

Figure 3: Illustrations of Lemma 2.

Proof. It is sufficient to show that for any three points x_1, x_2, x_3 appearing on W_i in that order from top to bottom, $\ell(x_1) = \ell(x_3)$ implies $\ell(x_2) < \ell(x_1) = \ell(x_3)$. Consider the shortest path from $f_{\mathcal{P}}(R_i)$ to $f_{\mathcal{Q}}(x_2)$, passing through R_i and W_i. Let x' be its intersection point with W_i. Then $\Pi(x')$ is the shortest path. Suppose that x_2 appears between x_1 and x'. The case of x_2 appearing between x' and x_3 can be similarly handled. Let Π be the path consisting of three segments $(f_{\mathcal{P}}(R_i), p(x_1))$, $(p(x_1), x_1)$ and $(x_1, f_{\mathcal{Q}}(x_2))$. Then, Π and $\Pi(x')$ form a polygon (see Figure 3(a)). In the polygon two edges $(f_{\mathcal{P}}(R_i), p(x_1))$ and $(f_{\mathcal{P}}(R_i), p(x'))$ may coincide (see Figure 3(b)). The polygon contains $\Pi(x_2)$. Since $\Pi(x_2)$ is a convex chain, its length, $\ell(x_2)$, is less than the length of Π, which is in turn less than $\ell(x_1)$. $\square$

Theorem 1 *Given two disjoint convex polygons $\mathcal{P}$ and $\mathcal{Q}$ of n and m vertices, we can compute the optimal bridge between $\mathcal{P}$ and $\mathcal{Q}$ in $O(n+m)$ time.*

3 Simple-convex problem

Let $\mathcal{P}$ be a non-convex simple polygon of n vertices and let $\mathcal{Q}$ be a convex polygon of m vertices. To find an optimal bridge $\mathcal{P}$ and $\mathcal{Q}$, as did in the convex-convex algorithm, we break up $\partial\mathcal{P}$ and $\partial\mathcal{Q}$ into line segments R_i's and L_j's such that an optimal bridge through R_i and L_j for any i, j can be computed in constant time. The difference is that an optimal bridge through each R_i and $\partial\mathcal{Q}$ is computed in $O(\log m)$ time by binary searches.

3.1 Restricted bridge problem

Before getting into the algorithm, we consider a simplified problem as follows: Given a point p outside a convex polygon $\mathcal{Q}$ of m vertices, find an optimal bridge of p and $\mathcal{Q}$ that minimizes

$$\ell(x) = d(p, x) + d(x, f_{\mathcal{Q}}(x))$$

for all points x which are visible from p. We will prove that this length function is *downward convex*, whose exact definition will be given below. This allows us to do binary search to find a minimum of the function in $O(\log m)$ time provided that $\mathrm{FVD}(\mathcal{Q})$ are already computed. This convexity will be used to figure out the property of the length function defined in the simple-convex problem.

As in Figure 4, define a visible portion L of $\partial\mathcal{Q}$ from p by computing two tangential points a, b at $\partial\mathcal{Q}$ from p. Assume without loss of generality that a is above b, and p is to the left of the line ab. As the next step, we partition L into line segments $L_j = (q_j, q_{j+1})$, where $q_1, \ldots, q_k$ is an ordered sequence (from top to bottom) of points in $\{\mathrm{FVD}(\mathcal{Q}) \cap L\} \cup V(L)$. Then $q_1 = a$ and $q_k = b$.

Since all points on L are visible from p, we can parameterize a point $x \in L$ by the angle between the lines pa and px. Thus each point $x \in L$ is associated with an angle θ_x. If γ_θ denotes a ray emanating from p to the direction $\theta_a \leq \theta \leq \theta_b$, then γ_θ intersects L exactly once at point x_θ. As a result, $\ell(x)$ is rewritten as a function of parameter θ (instead of parameter x) for $\theta_a \leq \theta \leq \theta_b$ as follows:

$$\ell(\theta) = d(p, x_\theta) + d(x_\theta, f_{\mathcal{Q}}(x_\theta)).$$

We now claim that $\ell(\theta)$ is a downward convex function, meaning that $h(\theta) > \ell(\theta)$ for any $\theta' < \theta < \theta''$, where h is a linear function which intersects $\ell(\theta)$ at θ' and θ''. Thus $\ell(\theta)$ can have (at most) one local minima. When the minimum occurs at $\theta = \theta^*$, the optimal bridge becomes (p, x_{θ^*}) and this can be computed in $O(\log m)$ time by binary search based on the convexity of $\ell(\theta)$.

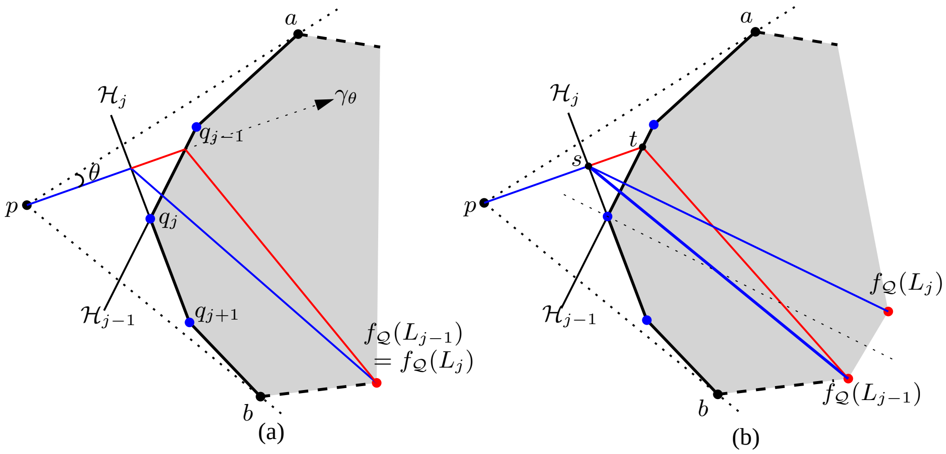

Figure 4: The proof of the convexity of $\ell(\theta)$.

Let $\mathcal{H}_j$ be a line extending $L_j = (q_j, q_{j+1})$ of L and let $\ell^j(\theta)$ for $\theta_{q_1} \leq \theta \leq \theta_{q_k}$ be the length of the path from p, passing through $\gamma_\theta \cap \mathcal{H}_j$, to $f_{\mathcal{Q}}(L_j)$. It is not difficult to show that $\ell^j(\theta)$ is downward convex. Thus, $\ell^j(\theta)$ is downward convex for $\theta_{q_j} \leq \theta \leq \theta_{q_{j+1}}$.

The function $\ell(\theta)$ for $\theta_{q_1} \leq \theta \leq \theta_{q_k}$ is a piecewise function consisting of $\ell^j(\theta)$ for $\theta_{q_j} \leq \theta \leq \theta_{q_{j+1}}$ for each $1 \leq j < k$. If there is a non-convex point on $\ell(\theta)$, then the point must occur at θ_{q_j} for some j. However, this is impossible because $\ell(\theta)$ for $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_{j+1}}$ for any $1 < j < k$ is downward convex as will be seen below.

Consider two consecutive segments L_{j-1} and L_j which share q_j as an end-point. We will prove that $\ell(\theta)$ for $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_{j+1}}$ is a maximum of $\ell^{j-1}(\theta)$ and $\ell^{j}(\theta)$, i.e.,

$$\ell(\theta) = \max_{\theta_{q_{j-1}} \leq \theta \leq \theta_{q_{j+1}}} \{\ell^{j-1}(\theta), \ell^{j}(\theta)\}.$$

Since both $\ell^{j-1}(\theta)$ and $\ell^{j}(\theta)$ are downward convex, their maximum is also downward convex.

We have two cases depending on whether $f_{\mathcal{Q}}(L_{j-1}) = f_{\mathcal{Q}}(L_j)$ or not. If they are same, then $q_j \in V(L)$. By the triangle inequality, as in Figure 4(a), $\ell^{j-1}(\theta) \geq \ell^{j}(\theta)$ for $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_j}$ and $\ell^{j-1}(\theta) \leq \ell^{j}(\theta)$ for $\theta_{q_j} \leq \theta \leq \theta_{q_{j+1}}$. Thus $\ell(\theta) = \max\{\ell^{j-1}(\theta), \ell^{j}(\theta)\}$ over $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_{j+1}}$.

If $f_{\mathcal{Q}}(L_{j-1}) \neq f_{\mathcal{Q}}(L_j)$, then q_j is a red point, which is the intersection of L with the bisector of the line segment connecting $f_{\mathcal{Q}}(L_{j-1})$ and $f_{\mathcal{Q}}(L_j)$. As in Figure 4(b), let $s = \mathcal{H}_j \cap \gamma_\theta$ and $t = \mathcal{H}_{j-1} \cap \gamma_\theta$ for a fixed $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_j}$. Since both s and t are above the bisector, their furthest point is $f_{\mathcal{Q}}(L_{j-1})$. Thus $\ell^{j}(\theta) = d(p,s) + d(s, f_{\mathcal{Q}}(L_j)) \leq d(p,s) + d(s, f_{\mathcal{Q}}(L_{j-1}))$. Since $d(s, f_{\mathcal{Q}}(L_{j-1})) < d(s,t) + d(t, f_{\mathcal{Q}}(L_{j-1}))$ by the triangle inequality, $\ell^{j}(\theta) < d(p,s) + d(s,t) + d(t, f_{\mathcal{Q}}(L_{j-1})) = \ell^{j-1}(\theta)$. Thus $\ell(\theta) = \ell^{j-1}(\theta) = \max\{\ell^{j-1}(\theta), \ell^{j}(\theta)\}$ for $\theta_{q_{j-1}} \leq \theta \leq \theta_{q_j}$. By a similar reasoning, $\ell(\theta) = \ell^{j}(\theta) = \max\{\ell^{j-1}(\theta), \ell^{j}(\theta)\}$ for $\theta_{q_j} \leq \theta \leq \theta_{q_{j+1}}$. As a result, we have the following result.

Lemma 3 *The length function $\ell(\theta)$ for $\theta_{q_1} \leq \theta \leq \theta_{q_k}$ is downward convex.*

Lemma 4 *Given a point p outside a convex polygon $\mathcal{Q}$ of m vertices, an optimal bridge of p and $\mathcal{Q}$ can be computed in $O(\log m)$ time if $\mathrm{FVD}(\mathcal{Q})$ and the partition of $\partial\mathcal{Q}$ (i.e., $q_1, \ldots, q_k$) are already computed in $O(m)$ time.*

Proof. The minimum of $\ell(\theta)$ is computed by binary search on the ordered sequence $\theta_{q_1}, \ldots, \theta_{q_k}$ as follows: (i) pick a middle segment $L_{\lfloor k/2 \rfloor} = (q_{\lfloor k/2 \rfloor}, q_{\lfloor k/2 \rfloor + 1})$, (ii) compute an angle θ^*, in constant time, at which $\ell^{\lfloor k/2 \rfloor}(\theta)$ is minimized, (iii) if $\theta^* = \theta_{q_{\lfloor k/2 \rfloor}}$, then the minimum of $\ell(\theta)$ occurs at $\theta_{q_1} \leq \theta \leq \theta_{q_{\lfloor k/2 \rfloor}}$, so remove the lower half $\{L_j \mid \lfloor k/2 \rfloor < j \leq k\}$ in the next recursive consideration; similarly, if $\theta^* = \theta_{q_{\lfloor k/2 \rfloor + 1}}$, then throw the upper half $\{L_j \mid 1 \leq j < \lfloor k/2 \rfloor\}$ away; otherwise, i.e., if $\theta_{q_{\lfloor k/2 \rfloor}} < \theta^* < \theta_{q_{\lfloor k/2 \rfloor + 1}}$, then the minimum of $\ell(\theta)$ occurs at θ^* due to the convexity of $\ell(\theta)$.
□

Remark. Actually, $\ell(\theta)$ is not necessarily required to be downward convex to find its minimum in $O(\log m)$ time by binary search. We can perform binary search even if $\ell(\theta)$ is unimodal. However, we need such convexity to prove the unimodality of the length function defined in the simple-convex problem. The property used is that the sum of two downward convex functions is downward convex, which is not necessarily true for unimodal functions.

3.2 Algorithm for the simple-convex problem

Consider $\mathrm{CH}(\mathcal{P} \cup \mathcal{Q})$, the convex hull of $\mathcal{P} \cup \mathcal{Q}$. This is computed in $O(n+m)$ time by classical algorithms [9]. Suppose for a while that $\partial\mathrm{CH}(\mathcal{P} \cup \mathcal{Q})$ contains at least one vertex of $\mathcal{Q}$, as in Figure 5(a); the other case, as illustrated in Figure 5(b), that $\mathcal{Q}$ is entirely contained inside $\mathrm{CH}(\mathcal{P})$ will be mentioned at the end of this section.

As shown in Figure 5(a), we define two edges e_l and e_r of $\partial\mathrm{CH}(\mathcal{P} \cup \mathcal{Q})$ which do not belong to $\partial\mathcal{P}$ and $\partial\mathcal{Q}$. Let R be the chain of $\partial\mathcal{P}$ which consists of edges that we meet as we traverse $\partial\mathcal{P}$ from the end-vertex of e_l to the end-vertex of e_r in clockwise order, and let L be the chain of $\partial\mathcal{Q}$ which consists of edges from the end-vertex of e_l to the end-vertex of e_r in counterclockwise order. Then any bridge of $\mathcal{P}$ and $\mathcal{Q}$ must have its two end-points on R and L which are visible each other.

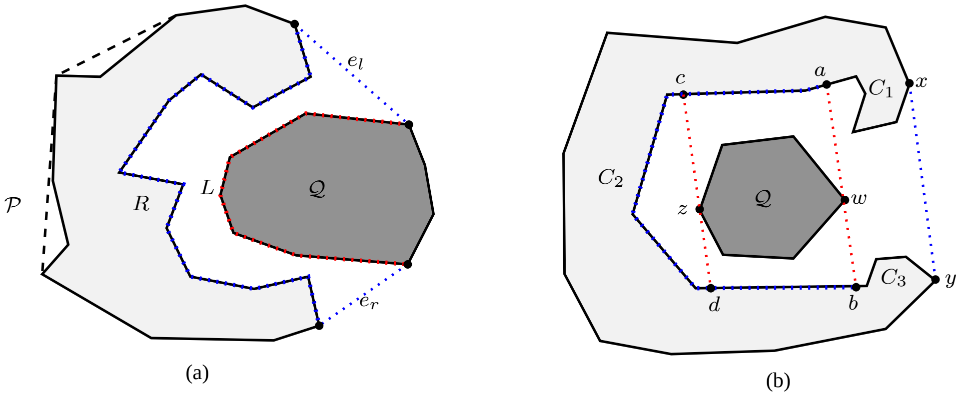

Figure 5: Two cases depending on whether or not $\mathcal{Q}$ is contained in the convex hull of $\mathcal{P}$.

Like the convex-convex algorithm in Section 2, we (i) partition R into smaller segments R_i, (ii) compute a weakly visible portion $W_i \subseteq L$ of R_i, and (iii) find an optimal bridge through R_i and W_i for each i. Before explaining it in detail, we need to introduce a refinement of furthest Voronoi diagrams in a simple polygon.

It is well known that for two points p, q in $\mathcal{P}$, a geodesic shortest path $\mathrm{SP}(p, q)$ from p to q in $\mathcal{P}$ is unique and passes through only reflex vertices of $\mathcal{P}$ except p and q. In particular, we call the first reflex vertex along $\mathrm{SP}(p, q)$ an *anchor vertex* (or simply, *anchor*) of p with respect to q. If the anchors of p and p' with respect to q are identical, then $\mathrm{SP}(p, q)$ and $\mathrm{SP}(p', q)$ pass through the same set of reflex vertices in the same order toward q. A *shortest path map* $\mathrm{SPM}(p)$ of a point $p \in \mathcal{P}$ is a decomposition of $\mathcal{P}$ into regions such that for all points q in a region have the same anchor with respect to p [6]. All points in a region in $\mathrm{SPM}(p)$ are visible from an anchor associated with the region. Moreover, a shortest path map in a simple polygon can be computed in linear time [6].

Aronov et al. [2] defined a *refined geodesic furthest Voronoi diagram* $\mathrm{FVD}^\star(\mathcal{P})$ of $\mathcal{P}$ so that all points p in a Voronoi region have the same geodesic furthest vertex $f_\mathcal{P}(p)$ of $\mathcal{P}$, and have the same anchor with respect to $f_\mathcal{P}(p)$.

Lemma 5 [2] *Given a set S of k points in a simple n-sided polygon $\mathcal{P}$, the partition of $\partial\mathcal{P}$ by a refined geodesic furthest Voronoi diagram of S in $\mathcal{P}$ has a complexity of $O(n + k)$ and is computed in $O((n + k)\log(n + k))$ time and $O(n + k)$ space.*

3.2.1 Partitioning R and L

We first compute $I_r = \{R \cap \mathrm{FVD}^\star(\mathcal{P})\}$, the *red* intersections of R with the Voronoi edges of $\mathrm{FVD}^\star(\mathcal{P})$. Next, number the points $I_r \cup V(R)$ as $p_1, p_2, \cdots, p_{n'}$ in clockwise order, where p_1 is an end-vertex of e_l and $p_{n'}$ is an end-vertex of e_r. Divide R into line segments $R_i = (p_i, p_{i+1})$ for $1 \le i < n'$. Then every point on R_i has the same anchor λ_i with respect to the same furthest vertex $f_\mathcal{P}(R_i)$. This takes $O(n \log n)$ time by Lemma 5.

We divide the chain L into line segments $L_j = (q_j, q_{j+1})$, where $q_1, q_2, \ldots, q_{m'}$ is a counterclockwise sequence of the points in $\{L \cap \mathrm{FVD}(\mathcal{Q})\} \cup V(L)$. Note here that we do not need to use

$\mathrm{FVD}^\star(\mathcal{Q})$ because $\mathcal{Q}$ is convex. This takes $O(m)$ time by Lemma 1.

Fix a segment R_i. The optimal bridge through R_i and L is a pair (p, q) of points minimizing

$$|\mathrm{SP}(f_{\mathcal{P}}(R_i), p)| + d(p, q) + d(q, f_{\mathcal{Q}}(q))$$

over all pairs of $p \in R_i$ and $q \in L$ such that p, q are visible each other. However, the shortest path $\mathrm{SP}(f_{\mathcal{P}}(R_i), p)$ always passes through the anchor vertex λ_i, thus the function is equal to

$$|\mathrm{SP}(f_{\mathcal{P}}(R_i), \lambda_i)| + d(\lambda_i, p) + d(p, q) + d(q, f_{\mathcal{Q}}(q)).$$

Notice here that p is visible from its anchor λ_i, so $|\mathrm{SP}(\lambda_i, p)|$ is replaced with $d(\lambda_i, p)$. The first term is a constant as long as $p \in R_i$, so it suffices to find a pair (p, q) minimizing

$$d(\lambda_i, p) + d(p, q) + d(q, f_{\mathcal{Q}}(q)).$$

We now define W_i for each R_i as a collection of points on L at which an end-point of a bridge through R_i and L can be located. Then W_i is a union of visible portions of L from each point on R_i. A point $q \in L$ is said to be *weakly visible* [3] from R_i if q is visible from some point on R_i, not necessarily from all points on R_i. Thus W_i is a weakly visible portion of L from R_i. Due to the convexity of L, W_i is connected. For an illustration, see Figure 6(a).

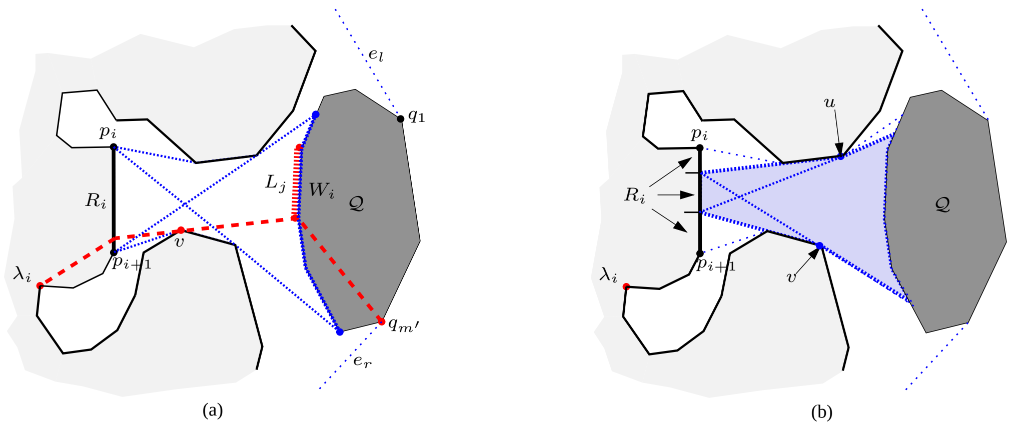

Figure 6: (a) Optimal path through R_i and L_j. (b) Splitting R_i into smaller segments.

To find an optimal bridge through R_i and W_i, we must compute an optimal bridge through R_i and L_j, where $L_j \cap W_i \neq \emptyset$, as fast as possible, e.g., in constant time. But this is not trivial, because some point on L_j may not be visible from every point on R_i. This was trivial in the convex-convex problem of Section 2, where all points on L_j for any $\alpha(i) \leq j < \beta(i)$ are completely visible from every point on R_i. As shown in Figure 6(a), an optimal bridge through R_i and L_j may touch $\mathcal{P}$ at some vertex v. Hence, we need to locate such vertices, if any. To guarantee constant cost, we further partition R_i into smaller segments so that there is a constant number of vertices at which all bridges with an end-point on a smaller segment can touch. For instance, in Figure 6(b), R_i is partitioned into three smaller segments, and all bridges associated with the middle segment are contained in the shaded region and may touch $\mathcal{P}$ at one of two vertices, u and v.

Let us explain how to divide R_i's further. We first define a new simple polygon $\mathcal{G}$ defined by a region between R and L. More precisely, the boundary of $\mathcal{G}$ consists of $e_l = (q_1, p_1)$, R, $e_r = (p_{n'}, q_{m'})$, and L in counterclockwise order (see Figure 5(a)). We next compute two shortest path maps, $\mathrm{SPM}(q_1)$ and $\mathrm{SPM}(q_{m'})$, in $\mathcal{G}$. Let I_b be the collection of *blue* intersections of R with edges of $\mathrm{SPM}(q_1)$ and $\mathrm{SPM}(q_{m'})$. Then all points between two consecutive blue intersections on R have the same anchor with respect to q_1 and the same anchor with respect to $q_{m'}$. Note that these anchors are defined in the new polygon $\mathcal{G}$. R_i's are further refined by adding these blue intersections.

We newly number the points in $I_b \cup I_r \cup V(R)$ as p_i's in clockwise order on $\partial\mathcal{P}$, and define segments $R_i = (p_i, p_{i+1})$ for each i. For the following notations, see Figure 7. Let u_i (resp., v_i) denote the anchor of R_i with respect to q_1 (resp., $q_{m'}$) in $\mathcal{G}$. Let a_i (resp., b_i) be the intersection of L with a line passing through p_{i+1} and u_i (resp., through p_i and v_i). If u_i (resp., v_i) is a vertex of L, then we set $a_i = u_i$ (resp., $b_i = v_i$). If u_i, v_i both are vertices of R and u_i is equal to v_i, then no points on R_i can see any point on L, so a_i and b_i are not defined. We, in the following, assume that u_i and v_i are distinct. Then W_i is a chain of L from a_i to b_i in top-to-bottom order along L (i.e., in counterclockwise order on $\partial\mathcal{Q}$).

Let us check the time complexity needed to do the above steps. Since a shortest path map in a simple polygon can be computed in linear time, we can compute $\mathrm{SPM}(q_1)$ and $\mathrm{SPM}(q_{m'})$ together with the blue intersections in $O(|\mathcal{G}|) = O(n+m)$ time. The number of R_i's created is also $O(n+m)$. Computing a_i and b_i is done by binary searches over L. Hence the total time to compute R_i's and W_i's is $O((n+m)\log m)$.

3.2.2 Finding an optimal bridge through R_i and W_i

For a point $x \in W_i$, let $p(x) \in R_i$ be an end-point of an optimal bridge whose other end-point is x, and let $\Pi(x)$ be a path from λ_i, containing the bridge $(p(x), x)$, to $f_{\mathcal{Q}}(x)$. Define $\ell(x)$ as a length of $\Pi(x)$, i.e.,

$$\ell(x) = d(\lambda_i, p(x)) + d(p(x), x) + d(x, f_{\mathcal{Q}}(x)).$$

We will show that $\ell(x)$ is unimodal, which means its minimum can be found in logarithmic time by binary search as in Lemma 3.

For the sake of the simplicity of the following description, we consider the most complicated configuration, that is, assume that u_i and v_i are all vertices of R. See Figure 7 for the following notations. Let c_i (resp., d_i) be the intersection of W_i with a line $p_i u_i$ (resp., a line $p_{i+1} v_i$). Clearly, four points, a_i, c_i, d_i, b_i, appear on L from top to bottom in that order. Let q (resp., q') be the intersection of W_i with a ray from λ_i, penetrating R_i, passing through u_i (resp., v_i). Note that q and q' may not exist. For example, in Figure 7, q' does not exist. Furthermore, if q (resp., q') exists, then it must lie between a_i and c_i (resp., between d_i and b_i). Thus there are only four cases according to whether q and q' exist. We will concentrate only on the case in which q exists but q' does not (see Figure 7). Applying similar arguments, we can easily handle the other cases, so we omit the details.

We define another intersection point r of W_i with a ray $\lambda_i p_{i+1}$. Six points, a_i, q, c_i, r, d_i, b_i, appear on L from top to bottom. We are here interested in the five points, excluding c_i. For convenience, we rename those five points as $x_1, \ldots, x_5$ from top to bottom.

Let γ_x denote a ray emanating from $p(x)$ to a point $x \in W_i$. Considering the behavior of $\Pi(x)$ for each subchain W_i^j, we can easily verify that for any two points $x, x' \in W_i$, γ_x and $\gamma_{x'}$ have no common intersection inside $\mathcal{Q}$. A point $x \in W_i$ is associated with an angle θ_x formed by γ_{x_1}

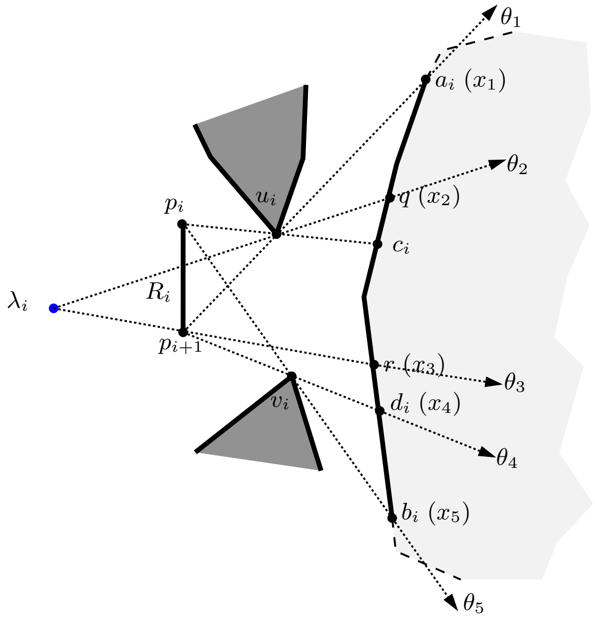

Figure 7: Definitions of a_i, q, c_i, r, d_i, b_i.

and γ_x. As x moves along W_i from x_1 to x_5, θ_x continuously changes from θ_{x_1} to θ_{x_5}. Thus, as in Section 3.1, we can parameterize $\ell(x)$ by angle θ_x, but we will use $\ell(x)$ instead of $\ell(\theta)$ for simplicity.

For a point $x \in W_i$, x is said to be *left-sided* if $f_{\mathcal{Q}}(x)$ is in the left side of a ray γ_x from $p(x)$ to x; *right-sided* if $f_{\mathcal{Q}}(x)$ is in the right side of γ_x; *both-sided* otherwise. Note that x may have two vertices of $\mathcal{Q}$ as $f_{\mathcal{Q}}(x)$, thus if x is both-sided, then one of two furthest vertices from x is in the left side of γ_x and the other is in the right side of γ_x.

We need several lemmas to prove the unimodality $\ell(x)$. We again assume that all points on W_i are ordered from top to bottom.

Lemma 6 *If $x \in W_i$ is left-sided, then any x' below x is left-sided. Symmetrically, if $x \in W_i$ is right-sided, then any x' above x is right-sided.*

Proof. We will prove only the first claim because the second is symmetric. For contradiction, suppose that there is a point x' below x such that x' is right-sided. Then we have that $f_{\mathcal{Q}}(x) \neq f_{\mathcal{Q}}(x')$. Since γ_x and $\gamma_{x'}$ have no common intersection in $\mathcal{Q}$, $f_{\mathcal{Q}}(x)$ must be above $\gamma_{x'}$ and $f_{\mathcal{Q}}(x')$ is below γ_x. Let h be a line bisecting the segment $(f_{\mathcal{Q}}(x), f_{\mathcal{Q}}(x'))$. Then x and $f_{\mathcal{Q}}(x)$ lie above h and x' and $f_{\mathcal{Q}}(x')$ lie below h. Since x and x' have different furthest vertices, h must intersect W_i at some point x'' which is below x and above x'. However this means that x is further to $f_{\mathcal{Q}}(x')$ than to $f_{\mathcal{Q}}(x)$ and x' is further to $f_{\mathcal{Q}}(x)$ than to $f_{\mathcal{Q}}(x')$. This is a contradiction. □

Corollary 1 *For some $x \in W_i$, if x is both-sided, then any x' below x is left-sided, and any x'' below x is right-sided. Moreover, if the both-sided point exists on W_i, then it is unique.*

We denote by W_i^j a subchain of W_i from x_j to x_{j+1} for $1 \leq j \leq 4$.

Lemma 7 *If x_2 is not right-sided, then $\ell(x)$ over $x \in W_i \setminus W_i^1$ is increasing. Symmetrically, if x_4 is not left-sided, then $\ell(x)$ over $x \in W_i \setminus W_i^4$ is decreasing.*

Proof. If x_2 is not right-sided, then any x below x_2 is left-sided by Lemma 6 and Corollary 1.

It suffices to show that $\ell(x)$ over each W_i^j for $2 \leq j \leq 4$ is increasing. To prove it, it suffices to show that if a is above b for any pair of points $a, b \in W_i^j$, then $\ell(a) < \ell(b)$.

Consider $\ell(x)$ over W_i^2. Every point on W_i^2 is visible from λ_i. For $a, b \in W_i^2$, we have two cases depending on whether $f_{\mathcal{Q}}(a) = f_{\mathcal{Q}}(b)$ or not. When $f_{\mathcal{Q}}(a) = f_{\mathcal{Q}}(b)$, a triangle $\lambda_i b f_{\mathcal{Q}}(b)$ contains another triangle $\lambda_i a f_{\mathcal{Q}}(a)$ as both a and b are left-sided. A simple geometric fact means that $\ell(a) < \ell(b)$. In the other case that $f_{\mathcal{Q}}(a) \neq f_{\mathcal{Q}}(b)$, it holds that $\ell(b) \geq d(\lambda_i, b) + d(b, f_{\mathcal{Q}}(a))$ since $f_{\mathcal{Q}}(b)$ is the furthest vertex from b. Consider two triangles $\lambda_i a f_{\mathcal{Q}}(a)$ and $\lambda_i b f_{\mathcal{Q}}(a)$ with a common base $(\lambda_i, f_{\mathcal{Q}}(a))$. Then $d(\lambda_i, b) + d(b, f_{\mathcal{Q}}(a)) > \ell(a)$, thus $\ell(b) > \ell(a)$. Thus $\ell(x)$ is increasing over W_i^2.

When $x \in W_i^3$, $\Pi(x)$ must pass the end-point p_{i+1} of R_i, so it is sufficient to prove the function $\ell'(x) = d(p_{i+1}, x) + d(x, f_{\mathcal{Q}}(x))$ is increasing. Since every point on W_i^3 is visible from p_{i+1}, we can apply the same argument used above to prove $\ell'(x)$ is increasing. Thus $\ell(x)$ is increasing over W_i^3.

When $x \in W_i^4$, then $\Pi(x)$ must pass the anchor v_i, so

$$\ell(x) = (d(\lambda_i, p(x)) + d(p(x), v_i)) + (d(v_i, x) + d(x, f_{\mathcal{Q}}(x))). \tag{1}$$

The first sum is clearly increasing. The second sum is shown to be increasing by the same argument used for the above cases. Thus their sum, $\ell(x)$, is increasing over W_i^4. As a result, $\ell(x)$ over each W_i^j $(2 \leq j \leq 4)$ is increasing. □

Lemma 8 *The length function $\ell(x)$ over W_i is unimodal.*

Proof. Assume that a (unique) both-sided point x^* exists on W_i; the proof in the absence of x^* is similar. Note that any x above x^* is right-sided and any x below x^* is left-sided by Corollary 1. Consider a case that $x^* \in W_i^2 \cup W_i^3$. Then, by Lemma 7, $\ell(x)$ from x_1 to x^* is decreasing and $\ell(x)$ over x from x^* to x_5 is increasing. Thus $\ell(x)$ is unimodal.

In the case that $x^* \in W_i^4$, x_4 is right-sided. Then $\ell(x)$ is represented as a sum of two sums as in Equation (1). Every point x on W_i^4 is visible from the anchor v_i, and every $p(x)$ on R_i is also visible from v_i, thus we can apply Lemma 3 to show that the first and second sum are all downward convex. Since the sum of two downward convex functions is still downward convex, $\ell(x)$ over W_i^4 is downward convex. Lemma 7 means that $\ell(x)$ over $W_i \setminus W_i^4$ is increasing. Hence $\ell(x)$ over W_i is unimodal

The case that $x^* \in W_i^1$ can be handled symmetrically. □

As a result, we have the following result.

Lemma 9 *For some R_i, the minimum of $\ell(x)$ for $x \in W_i$ can be found in $O(\log m)$ time.*

Considering all R_i's, we can compute the optimal bridge through R and L in $O((n+m)\log m)$ time. Since $\mathrm{FVD}^\star(\mathcal{P})$ is obtained in $O(n \log n)$ time, the first case of Figure 5(a) is solved in

$O((n+m)\log(n+m))$ time.

Finally, we have to handle the second case, as illustrated in Figure 5(b), that $\mathcal{Q}$ is entirely contained in the convex hull of $\mathcal{P}$. We will explain how to transform this into two sub-problems of the first case. Suppose that, as in Figure 5(b), $\mathcal{Q}$ is contained in a region which are bounded by a non-polygonal edge (x, y) and a chain C of $\partial\mathcal{P}$ from x to y in clockwise order. Find a vertex w of $\mathcal{Q}$ at which a line ℓ parallel to (x, y) touches $\mathcal{Q}$. Among the intersections of ℓ with C, denote two intersections by a and b, and assume that a is above b. Then a and b divide C into three chains C_i for $i = 1, 2, 3$, as illustrated in Figure 5(b); C_1 from x to a, C_2 from a to b, and C_3 from b to y in clockwise order.

Suppose that an end-point of a bridge is on the middle chain C_2. Then the other point must be on the chain E of $\partial\mathcal{Q}$ which consists of all vertices on $\partial\mathcal{Q}$; if we think of w as a degenerated edge, then the chain E has w its two end-points. We can compute an optimal bridge between C_2 and E by the method of the first case; here C_2, E, (a, w), (w, b) are regarded as R, L, e_l, e_r, respectively.

Suppose now that an end-point of a bridge is either on C_1 or on C_3. We will make $\partial\mathcal{Q}$ a chain by cutting it at some vertex, and use it as chain on which the other end-point lies. Let ℓ' be a line parallel to ℓ and touching $\partial\mathcal{Q}$ at the point z. If we also think of z as a degenerated edge, a chain containing all vertices of $\mathcal{Q}$ is defined. Let c and d be the intersections of ℓ' with C_2 as in Figure 5(b). Now regard as R a chain that we meet as we traverse the boundary $C \cup (x, y)$ from c to d in clockwise order; as L the chain $\partial\mathcal{Q}$ with z as its two end-points; here e_l and e_r are (c, z) and (d, z), respectively. Since an end-point of bridges cannot be on (x, y), we just ignore the bridges with one of their end-points on (x, y) during the execution of the algorithm. Hence, we have the following result.

Theorem 2 *Given a non-convex simple polygon $\mathcal{P}$ of n vertices and a convex polygon $\mathcal{Q}$ of m vertices, we can compute the optimal bridge between $\mathcal{P}$ and $\mathcal{Q}$ in $O((n+m)\log(n+m))$ time.*

4 Simple-simple problem

Let $\mathcal{P}$ and $\mathcal{Q}$ be non-convex simple polygons of n and m vertices, respectively. The optimal bridge of $\mathcal{P}$ and $\mathcal{Q}$ can be computed in $O((n+m)^2)$ time in a straightforward manner as follows: after partitioning $\partial\mathcal{P}$ and $\partial\mathcal{Q}$ into $O(n+m)$ line segments such that the optimal bridge over each pair of the segments can be computed in constant time, just examine all $O((n+m)^2)$ pairs of the segments and take a minimum among them. In this section, we present another algorithm which uses the simple-convex algorithm presented in the previous section as a subroutine and runs in $O(nm + m\log m)$ time if $m \geq n$.

After dividing the problem into two cases like Figure 5 in Section 3, we solve the first case where $\mathrm{CH}(\mathcal{P} \cup \mathcal{Q})$ contains at least one vertex of $\mathcal{Q}$ on its boundary.

For $R \subseteq \partial\mathcal{P}$ and $L \subseteq \partial\mathcal{Q}$ on which a bridge can be placed, we divide R into a collection of segments R_i's by $\{R \cap \mathrm{FVD}^\star(\mathcal{P})\} \cup V(R)$; similarly divide L into L_j's. The running time of this step is dominated by the time of computing $\mathrm{FVD}^\star(\cdot)$, so it becomes $O(n\log n + m\log m)$ by Lemma 5.

We will find an optimal bridge through R_i and L for each i in $O(n+m)$ time by using the simple-convex algorithm. To do it, like the simple-convex algorithm, define a new polygon $\mathcal{G}$ whose boundary consists of R, L, and two non-polygonal edges e_l and e_r that connect R and L. Consider a chain C, $\partial\mathcal{G} \setminus R_i$, i.e., a chain of $\partial\mathcal{G}$ obtained by deleting R_i from $\partial\mathcal{G}$. Then C consists of $R \setminus R_i$, e_l, L, and e_r. Let us now apply the simple-convex algorithm to find an optimal bridge between R_i and L. We here switch the roles of R_i and L each other, that is, run the simple-convex algorithm with two chains C (as a boundary part R of a simple polygon) and R_i (as a boundary part L of a convex polygon). Of course, during the execution of the algorithm, we may encounter steps to compute infeasible bridges whose end-points are on $R \setminus R_i$ and R_i, respectively. This is simply overcome by just ignoring such bridges when they are considered. Therefore, by Theorem 2, we can find an optimal bridge between R_i and L in $O(|\mathcal{G}|) = O(n+m)$ time. (Note that the $O(\log m)$-factor in the running time of Theorem 2 comes from the binary search over a convex boundary portion of size $O(m)$. We do not need it here because the convex boundary part consists of a single segment R_i, so the time bound is dominated only by the size of $\mathcal{G}$.) Since there are $O(n)$ R_i's, the total time becomes $O((n+m)n)$ plus the cost to compute $\mathrm{FVD}^\star(\mathcal{P})$ and $\mathrm{FVD}^\star(\mathcal{Q})$. Thus we have the following result.

Theorem 3 *Given two disjoint simple polygons $\mathcal{P}$ and $\mathcal{Q}$, we can compute the optimal bridge between $\mathcal{P}$ and $\mathcal{Q}$ in $O(nm + m\log m)$ time, where $n = |\mathcal{P}|$, $m = |\mathcal{Q}|$, and $m \geq n$.*

5 Concluding remarks

We investigated the problems to connect two polygons $\mathcal{P}$ and $\mathcal{Q}$ by an optimal bridge (line segment) with end-points on $\partial\mathcal{P}$ and $\partial\mathcal{Q}$ such that the length of the longest path to connect two points in $\mathcal{P}$ and q through the bridge is minimized.

We can extend the bridge problems into ones in more general environment as follows: Given a set S of points in $\mathcal{P}$ and a set T of points in $\mathcal{Q}$, find a bridge (p,q) for $p \in \partial\mathcal{P}$ and $q \in \partial\mathcal{Q}$ which are visible each other such that minimizes $\max_{s\in S}(|\mathrm{SP}(s,p)|) + d(p,q) + \max_{t\in T}(|\mathrm{SP}(q,t)|)$. Since the lemmas developed in previous section are still valid even in the general problems, we can solve them in the same time bound after computing (refined) furthest Voronoi diagram of S and T in $O(n\log n + m\log m)$ time.

Theorem 4 *In the general problem setting, where $n = |\mathcal{P}| + |S|$ and $m = |\mathcal{Q}| + |T|$, if one of $\mathcal{P}$ and $\mathcal{Q}$ is non-convex, then an optimal bridge can be found in $O((n+m)\log(n+m))$ time and if both are non-convex, then in $O(nm + m\log m)$ time for $m \geq n$.*

We can easily compute a 2-approximate bridge in linear time, which means the length of the longest path through the bridge is not larger than twice the optimal path length. Such approximation algorithm directly comes from the method [5] used to find a 2-approximate bridge for two convex polygons. In [5], it was shown that the bridge connecting the nearest visible points of two convex polygons guarantees a 2-approximation. The method can be easily applied to the cases that the polygons are non-convex. However, it remains open to achieve the better approximation, especially on the simple-simple problem.

Other interesting open problems are to consider the 3D counterparts of the 2D bridge problems. A problem which seems tractable among them is to find an optimal bridge between two convex polyhedra. We think it is not trivial to solve the problem even within a quadratic time.

References

[1] A. Aggarwal, L. J. Guibas, J. Saxe, and P. W. Shor. A linear-time algorithm for computing the Voronoi diagram of a convex polygon. *Discrete Comput. Geom.*, 4(6):591–604, 1989.

[2] B. Aronov, S. J. Fortune, and G. Wilfong. Furthest-site geodesic Voronoi diagram. *Discrete Comput. Geom.*, 9:217–255, 1993.

[3] D. Avis, T. Gum, and G. Toussaint. Visibility between two edges of a simple polygon. *Visual Comput.*, 2(6):342–357, December 1986.

[4] P. Bose, F. Hurtado-Diaz., E. Omaña-Pulido, and G. Toussaint. Aperture angle optimization problems. In *Proc. 7th Canad. Conf. Comput. Geom.*, pages 73–78, 1995.

[5] L. Cai, Y. Xu, and B. Zhu. Computing the optimal bridge between two convex polygons. *Information Processing Letters*, 69:127–130, 1999.

[6] L. J. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. E. Tarjan. Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. *Algorithmica*, 2:209–233, 1987.

[7] B.-J. Kim, C.-S. Shin, and K.-Y. Chwa. Linear algorithms for computing a variant segment center. In *Proc. in KISS Conf.*, volume A, Spring, pages 708–710, (in Korean), 1998.

[8] S. K. Kim. The canal problems. *Journal of the Korea Information Science Society*, 20(12):1891–1901, (in Korean), 1993.

[9] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.