An Efficient Algorithm for Computing High-Quality Paths amid Polygonal Obstacles*

Pankaj K. Agarwal[1], Kyle Fox[1], and Oren Salzman[2]

[1]Duke University, Durham, North Carolina, USA
[2]Tel Aviv University, Tel Aviv, Israel

Abstract

We study a path-planning problem amid a set $\mathcal{O}$ of obstacles in $\mathbb{R}^2$, in which we wish to compute a short path between two points while also maintaining a high clearance from $\mathcal{O}$; the clearance of a point is its distance from a nearest obstacle in $\mathcal{O}$. Specifically, the problem asks for a path minimizing the reciprocal of the clearance integrated over the length of the path. We present the first polynomial-time approximation scheme for this problem. Let n be the total number of obstacle vertices and let $\varepsilon \in (0, 1]$. Our algorithm computes in time $O(\frac{n^2}{\varepsilon^2} \log \frac{n}{\varepsilon})$ a path of total cost at most $(1 + \varepsilon)$ times the cost of the optimal path.

1 Introduction

Motivation. Robot motion planning deals with planning a collision-free path for a moving creature in an environment cluttered with obstacles [6]. It has applications in diverse domains such as surgical planning and computational biology. Typically, a *high-quality* path is desired where quality can be measured in terms of path length, clearance (distance from nearest obstacle at any given time), or smoothness, to mention a few criteria.

Problem statement. Let $\mathcal{O}$ be a set of polygonal obstacles in the plane, consisting of n vertices in total. A path γ for a point robot moving in the plane is the image of a continuous function $\gamma : [0, 1] \to \mathbb{R}^2$. Let $\|p, q\|$ denote the Euclidean distance between two points p, q. The *clearance* of a point p, denoted by $\text{cl}(p) := \min_{o \in \mathcal{O}} \|p, o\|$, is the minimal Euclidean distance between p and an obstacle ($\text{cl}(p) = 0$ if p lies in an obstacle). Similarly, the clearance of a path is defined as $\text{cl}(\gamma) := \min_{\tau \in [0,1]} \text{cl}(\gamma(\tau))$.

We use the following cost function, as defined by Wein *et al.* [17], that takes both the length and the clearance of a path γ into account:

$$\mu(\gamma) := \int_\gamma \frac{1}{\text{cl}(\gamma(\tau))} \text{d}\tau \tag{1.1}$$

This criteria is useful in many situations because we wish to find a short path that does not pass too close to the obstacles because of safety requirements. We abuse notation and let $\mu(p, q)$ be the minimal cost[1] of any path between p and q.

The *(approximate) minimal-cost path problem* is defined as follows: Given the set of obstacles $\mathcal{O}$ in $\mathbb{R}^2$, a real number $\varepsilon \in (0, 1]$, a start position s and a target position t, compute a path between s and t with cost at most $(1 + \varepsilon) \cdot \mu(s, t)$.

Related work. There is extensive work in computational geometry on computing shortest collision-free paths for a point moving amid a set of planar obstacles, and by now optimal $O(n \log n)$ algorithms are known; see Mitchell [12] for a survey and [5, 10] for recent results. There is also work on computing paths with the minimum number of links [13]. A drawback of these paths is that they may touch obstacle boundaries and therefore their clearance may be zero. Conversely, if maximizing the distance from the obstacles is the optimization criteria, then the path can be computed by constructing a maximum spanning tree in the Voronoi diagram of the obstacles (see Ó'Dúnlaing and Yap [14]). Wein *et al.* [16] considered the problem of computing shortest paths that have clearance at least δ for some parameter δ. However, this measure does not quantify the tradeoff between the length and the clearance, so Wein *et al.* [17] suggested the cost function defined in (1.1)

*Work by P.A. and K.F. has been supported in part by NSF under grants CCF-09-40671, CCF-10-12254, CCF-11-61359, IIS-14-08846, and CCF-15-13816, and by Grant 2012/229 from the U.S.-Israel Binational Science Foundation. Work by O.S. has been supported in part by the Israel Science Foundation (grant no.1102/11), by the German-Israeli Foundation (grant no. 1150-82.6/2011), and by the Hermann Minkowski–Minerva Center for Geometry at Tel Aviv University.

[1]Wein *et al.* assume the minimal-cost path exists. One can formally prove its existence by taking the limit of paths approaching the infimum cost.

to balance between minimizing the path length while maximizing its clearance. They devise an approximation algorithm to compute near-optimal paths under this metric for a point robot moving amidst polygonal obstacles in the plane. Their approximation algorithm runs in time polynomial in $\frac{1}{\varepsilon}, n$ and Λ where ε is the maximal additive error, n is the number of obstacle vertices and Λ is (roughly speaking) the total cost of the edges in the Voronoi diagram of the obstacles[2]. We are not aware of any polynomial-time approximation algorithm for this problem. It is not known whether the problem of computing the optimal path is NP-hard.

The problem of computing shortest paths amid polyhedral obstacles in $\mathbb{R}^3$ is NP-Hard [3], and a few heuristics have been proposed in the context of sampling-based motion planning in high dimensions (a widely used approach in practice [6]) to compute a short path that has some clearance; see, e.g., [15].

Several other bicriteria measures have been proposed in the context of path planning amid obstacles in $\mathbb{R}^2$, which combine the length of the path with curvature, the number of links in the path, the visibility of the path, etc. (see e.g. [1, 4, 11] and references therein). We also note a recent work, which is dual to the problem studied here [7]: Given a point set P and a path γ, they define the cost of γ to be the integral of clearance along the path, and the goal is to compute a minimum-cost path between two given points. They present an approximation algorithm whose running time is near-linear in the number of points.

Our Contribution. We present an algorithm[3] that given $\mathcal{O}, s, t$ and $\varepsilon \in (0, 1]$, computes in time $O\left(\frac{n^2}{\varepsilon^2} \log \frac{n}{\varepsilon}\right)$ a path from s to t whose cost is at most $(1 + \varepsilon) \cdot \mu(s, t)$.

As in [17], our algorithm is based on sampling, i.e., it contains a weighted geometric graph $G = (V, E)$ with $V \subset \mathbb{R}^2$ and $s, t \in V$ and computes a minimum-cost path in G from s to t. However, we prove a number of useful properties of optimal paths to obtain a fast algorithm.

We first refine the Voronoi diagram $\mathcal{V}$ of $\mathcal{O}$ into constant-size cells, which we refer to as the *refined Voronoi diagram* of $\mathcal{O}$ and denote it by $\tilde{\mathcal{V}}$. We prove (in Section 3) the existence of a path γ' from s to t whose cost is $O(\mu(s, t))$ and that has the following useful properties: (i) for every cell $T \in \tilde{\mathcal{V}}$, $\gamma' \cap \text{int}(T)$ is a connected subpath and the clearance of all points in this subpath is the same; (ii) for every edge $e \in \tilde{\mathcal{V}}$, there are $O(1)$ points, called *anchor points*, that depend only on the two cells incident to e with the property that either γ' intersects e transversally (i.e., $\gamma' \cap e$ is a single point) or the endpoints of the closure of γ' intersect e at anchor points. We say γ' consists of *well-behaved* paths. We use anchor points to propose a simple $O(n)$-approximation algorithm (Section 4.1), which we then transform into an $O(1)$-approximation algorithm (Section 4.2). We also use anchor points and the existence of well-behaved paths to choose a set of $O(n)$ or $O(n/\varepsilon)$ sample points on each edge of $\tilde{\mathcal{V}}$ with a total of $O(n^2)$ or $O(n^2/\varepsilon)$ samples in total (Sections 4.2 and 4.3).

We prove additional properties of optimal paths to construct the final graph with $O((n^2/\varepsilon^2) \log n/\varepsilon)$ edges (Section 4.3) instead of connecting every pair of sample points by an edge. Roughly speaking, we show that one can construct a small-size spanner.

[2]For the exact definition of Λ, see [17].

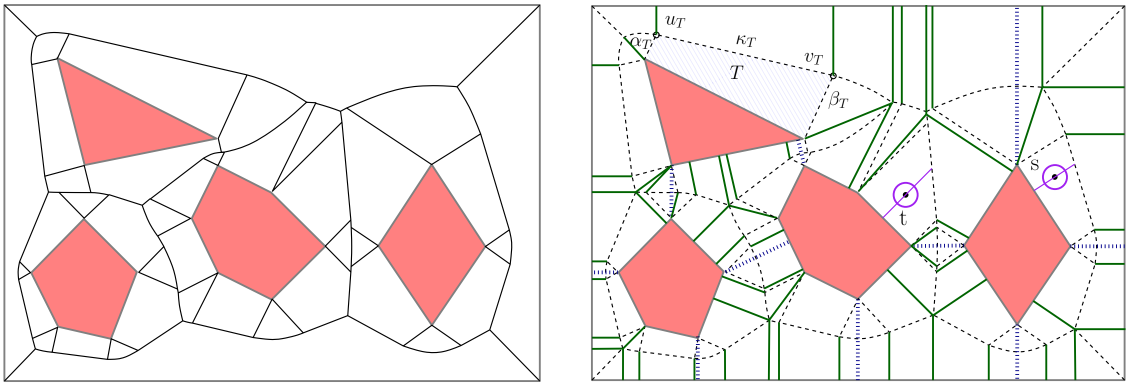
[3]We assume in this paper that $\mathbb{R}^2 \setminus (\bigcup \mathcal{O})$ is bounded. Our algorithm can be modified easily to avoid this assumption.

2 Preliminaries

Recall that $\mathcal{O}$ is a set of polygonal obstacles in the plane consisting of n vertices in total. We refer to the edges and vertices of $\mathcal{O}$ as its *features*. Given a point p and a feature o, let $\psi_o(p)$ be the closest point to p on o and $\|p, o\| = \|p, \psi_o(p)\|$. If a path γ contains two points p and q, we let $\gamma[p, q]$ denote the subpath of γ between p and q.

Voronoi diagram and its refinement. The *Voronoi cell* of a polygon feature o is the set of points in the closure of $\mathbb{R}^2 \setminus (\bigcup \mathcal{O})$ whose distance to any feature in $\mathcal{O}$ is minimized by o. The Voronoi cells of features are connected and internally disjoint. The Voronoi diagram $\mathcal{V}$ of features of $\mathcal{O}$ is the planar subdivision of the closure of $\mathbb{R}^2 \setminus (\bigcup \mathcal{O})$ determined by the Voronoi cells of features in $\mathcal{O}$. Voronoi edges between a line and a point obstacle's cells are parabolic arcs, and Voronoi edges between two line obstacles' or two point obstacles' cells are line segments. See [2] for details.

Note that for any obstacle feature o, and for any point x along any Voronoi edge on the boundary of o's Voronoi cell the function $\|x, \psi_o(x)\|$ is convex. We define the *refined Voronoi diagram* $\tilde{\mathcal{V}}$ by adding the following edges to $\mathcal{V}$: (i) the line segments $x\psi_o(x)$ between each obstacle feature o and Voronoi vertex x on the boundary of o's Voronoi cell, (ii) the line segment $x\psi_o(x)$ between each obstacle feature o and the point x along each Voronoi edge bounding o's cell that minimizes $\|x, \psi_o(x)\|$, and (iii) a line segment from the obstacle feature o closest to s (to t) that initially follows $\phi_o(s)s$ (or $\phi_o(t)t$) and ends at the first Voronoi edge it intersects. We refer to these extra edges as type (i), type (ii), or type (iii) edges respectively. Note that some type (i) edges may already be present in the Voronoi diagram $\mathcal{V}$. We say that an edge in $\tilde{\mathcal{V}}$ is an *internal edge* if it separates



(a) Voronoi Diagram

(b) Refined Voronoi Diagram

Figure 1: Voronoi diagram and Refined Voronoi diagram of a set of obstacles (dark red). (a) Voronoi edges depicted by solid black lines. (b) Voronoi edges depicted by dashed black lines, Green solid lines and blue dotted lines represent type (i) and type (ii) edges, respectively. Lines through s and t represent type (iii) edges. A representative cell T depicted in light blue.

two cells incident to the same polygon. Other edges are called *external edges*.

Clearly, the complexity of $\tilde{\mathcal{V}}$ is $O(n)$. Moreover, each cell T in $\tilde{\mathcal{V}}$ is incident to a single obstacle feature and has three additional edges. One edge (an external edge) of T is a monotone parabolic arc (we view a line segment as a parabolic arc); it is incident to two internal edges on T. For each cell T, let κ_T be the external edge of T, let α_T and β_T be the shorter and longer internal edges of T, respectively, and let u_T and v_T be the vertices connecting α_T and β_T to κ_T respectively. See Figure 1b.

Properties of optimal paths. We list several properties of our cost function. For detailed explanations and proofs, the reader is referred to Wein *et al.* [17]. Let $s = r_s e^{i\theta_s}$ be a start position and $t = r_t e^{i\theta_t}$ be a target position.

- Let o be a point obstacle with $\mathcal{O} = \{o\}$, and assume without loss of generality that o lies at the origin and $0 \le \theta_s \le \theta_t \le \pi$. The optimal path between s and t (see Figure 2a) is a logarithmic spiral centered on o, and its cost is

$$\mu(s,t) = \sqrt{(\theta_t - \theta_s)^2 + (\ln r_t - \ln r_s)^2}. \quad (2.2)$$

- Let o be a line obstacle with $\mathcal{O} = \{o\}$, and assume without loss of generality that o is supported by the line $y = 0$ and $0 \le \theta_s \le \theta_t \le \pi$. The optimal path between s and t (see Figure 2b) is a circular arc with its center at the origin, and its cost is[4]

$$\mu(s,t) = \ln \frac{1-\cos\theta_t}{\sin\theta_t} - \ln \frac{1-\cos\theta_s}{\sin\theta_s} = \ln\tan\frac{\theta_t}{2} - \ln\tan\frac{\theta_s}{2}. \quad (2.3)$$

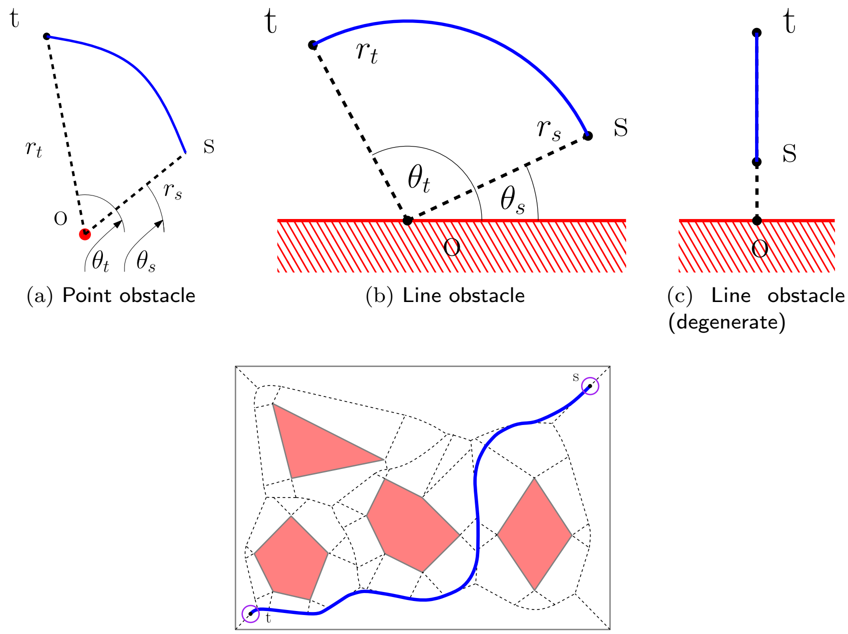
- Let o be an obstacle with $\mathcal{O} = \{o\}$ and s on the line segment between $\psi_o(t)$ and t. The optimal path between s and t (see Figure 2c) is a line segment, and its cost is

$$\mu(s,t) = \ln \mathrm{cl}(t) - \ln \mathrm{cl}(s). \quad (2.4)$$

- The minimal-cost path γ between two points p and q on an edge e of $\mathcal{V}$ is the piece of e between p and q. Moreover, there is a closed-form formula describing the cost of γ. Therefore, since each point within a single Voronoi cell is closest to exactly one obstacle feature...
- Given a set of obstacles, the optimal path connecting s and t consists of a sequence of circular arcs, pieces of logarithmic spirals, line segments, and pieces of Voronoi edges (see Figure 2d).

The following Corollary follows immediately from (1.1) and (2.2).

[4] The original equation describing the cost of the optimal path in the vicinity of a line obstacle had the obstacle on $x = 0$ and contained a minor inaccuracy. We present the correct cost in (2.3).



(a) Point obstacle

(b) Line obstacle

(c) Line obstacle (degenerate)

(d) Polygonal obstacle

Figure 2: Different examples of optimal paths (blue) among different types obstacles (red). In (d) the Voronoi diagram of the obstacles is depicted by solid black lines.

Corollary 2.1. *Let p and q be two points such that $\mathrm{cl}(p) \le \mathrm{cl}(q)$. The following properties hold:*
(i) A lower-bound on the cost of the shortest path between any p, q as defined above is $\mu(p,q) \ge \ln \frac{\mathrm{cl}(q)}{\mathrm{cl}(p)}$. If they lie in the same Voronoi cell of an obstacle feature o and if p lies on the minimal-cost path from o to q then the bound is tight.
(ii) Given a single point obstacle o located at the origin, $p = r_p e^{i\theta_p}$ and $q = r_q e^{i\theta_q}$ with $0 \le \theta_p \le \theta_q \le \pi$, a lower-bound on the cost of the shortest path between p, q is $\mu(p,q) \ge \theta_q - \theta_p$. If $r_p = r_q$ (namely, they are equidistant to o) then the bound is tight.

3 Well-behaved Paths

Let T be a cell of $\tilde{\mathcal{V}}$ incident to obstacle feature o, and let p and q be two points on the edges of T. Let γ be any path from p to q that does not leave T. We say γ is a *well-behaved* path if (i) $\gamma \cap \mathrm{int}(T)$ is a connected subpath and (ii) if it exists, then $\gamma \cap \mathrm{int}(T)$ has constant clearance. For a well-behaved path γ, let $\lambda(\gamma) = \gamma \cap \mathrm{int}(T)$. We often use λ in place of $\lambda(\gamma)$ when γ is clear from context. If o is a vertex, then λ is a circular arc centered at o, and if o is an edge, then λ is a line segment parallel to o. We have the following lemma.

Lemma 3.1. *Let T be a cell of $\tilde{\mathcal{V}}$, and let p and q be two points on the edges of T. There exists a well-behaved (p,q)-path γ within T where $\mu(\gamma) \le 7\mu(p,q)$.*

Proof. Let o be the obstacle feature incident to T. Let $\mathrm{cl}_{\max}(p,q)$ be the maximum clearance achieved by the minimal-cost path between p and q. Let T' be the subset of T restricted to points of clearance at most $\mathrm{cl}_{\max}(p,q)$. Path γ follows the unique path from p to q along the boundary of T' that does not intersect o. In particular, if both p and q lie on the same edge of T, then γ is the minimal-cost path from p to q (Corollary 2.1). We have three more cases to consider (and their symmetries).

Case 1) Points p and q lie on α_T and κ_T respectively. Path γ travels along α_T from p to u_T, and then along κ_T from u_T to q. By Corollary 2.1 and the fact that u_T is the lowest clearance point on κ_T relative to p, we have $\mu(p, u_T) \le \mu(p,q)$. By the triangle inequality we have that $\mu(u_T, q) \le \mu(u_T, p) + \mu(p,q) < 2\mu(p,q)$. Finally, $\mu(\gamma) \le \mu(p, u_T) + \mu(u_T, q) \le 3\mu(p,q)$. See Figure 3a.

Case 2) Points p and q lie on β_T and κ_T respectively. Let w be the endpoint of $\lambda = \lambda(\gamma)$ on β_T and let w' be the endpoint lying on κ_T. Path γ travels along β_T from p to w, along λ, and then along κ_T from w' to q.

Again, $\mu(p,w) \le \mu(p,q)$. If the obstacle defining T is a polygon edge, then λ is the Euclidean shortest path between any pair of points on β_T and κ_T whose clearance never exceeds $\mathrm{cl}_{\max}(p,q)$. It also (trivially) has the highest clearance of any such path. If the obstacle is a polygon vertex, then λ spans a shorter angle relative to the vertex than any other path whose clearance never exceeds $\mathrm{cl}_{\max}(p,q)$. By Corollary 2.1, the cost of any path from β_T to κ_T is at least this angle, and by (2.2), the cost of λ is exactly this lower bound. Either way, any path between p and q also goes between β_T and κ_T, so we conclude that $\mu(\lambda) \le \mu(p,q)$. We have $\mu(w',q) \le \mu(\lambda) + \mu(w,p) + \mu(p,q) \le 3\mu(p,q)$. Therefore, $\mu(\gamma) \le \mu(p,w) + \mu(\lambda) + \mu(w',q) \le 5\mu(p,q)$. See Figure 3b.

Case 3) Points p and q lie on β_T and α_T respectively. Let w be the endpoint of $\lambda = \lambda(\gamma)$ on β_T and let w' be the other endpoint. As before, $\mu(p,w) \le \mu(p,q)$ and $\mu(\lambda) \le \mu(p,q)$.

Suppose w' is on κ_T. Path γ travels along β_T from p to w, along λ, along κ_T to u_T, and then along α_T to q. We have $\mathrm{cl}_{\max}(p,q) \ge \mathrm{cl}(w') \ge \mathrm{cl}(u_T)$, so $\mu(q,u_T) \le \mu(p,q)$. Therefore $\mu(w',u_T) \le \mu(\lambda) + \mu(w,p) + \mu(p,q) + \mu(q,u_T) \le 4\mu(p,q)$. Finally $\mu(\gamma) = \mu(p,w) + \mu(\lambda) + \mu(w',u_T) + \mu(u_T,q) \le 7\mu(p,q)$. See Figure 3c.

Now, suppose w' is on α_T. Path γ travels along β_T from p to w, along λ, and then along α_T from w' to q. We have $\mu(q,w') \le \mu(p,q)$. Therefore, $\mu(\gamma) \le \mu(p,w) + \mu(\lambda) + \mu(w',q) \le 3\mu(p,q)$. See Figure 3d.

In the proof of Lemma 3.1, we chose subpath $\lambda = \lambda(\gamma)$ based on the maximum clearance of the minimal-cost (p,q)-path. Given λ and a point p on β_T, let $\gamma(p,\lambda)$ be the path that walks along β_T from p to λ and then walks along λ. We argued that $\mu(\gamma(p,\lambda)) \le O(\mu(p,q))$. Point w on β_T was the endpoint of λ. In the following lemma, we prove the existence of two *anchor points* w^*_α and w^*_κ on β_T which help us pick a suitable λ without knowing anything about the minimal-cost (p,q)-path other than its endpoint p (note that λ does not exist when neither p nor q use edge β_T). As we show, the anchor points can be computed in constant time given T.

LEMMA 3.2. *Let T be a cell of $\tilde{\mathcal{V}}$. There exist points w^*_α and w^*_κ on β_T such that the following holds: Let p and q be two points on the edges of T. There exists a well-behaved (p,q)-path γ within T such that $\mu(\gamma) \le 7\mu(p,q)$. If neither p nor q lie on β_T, then γ stays on α_T and κ_T. Otherwise $\lambda(\gamma) \cap \beta_T \in \{w^*_\alpha, w^*_\kappa, q, p\}$, and γ avoids at least one of α_T or κ_T.*

Proof. Let o be the obstacle feature incident to T. We assume at least one of p and q lie on β_T. Otherwise, we simply use the well-behaved path that follows one or both of α_T and κ_T. See the proof of Lemma 3.1.

Without loss of generality, p lies on β_T. We will pick $\lambda = \lambda(\gamma)$ so that we minimize the cost of $\gamma(p,\lambda)$. The lemma follows then by using our choice of λ in the proof of Lemma 3.1. Observe that for λ that minimizes the cost of $\gamma(p,\lambda)$, the endpoint of λ on β_T never lies closer to o than p; the value $\mu(\lambda)$ cannot decrease as one moves λ below p.We will now consider several cases based on the shape of T's edges and the edge containing q.

Case 1) Suppose o is a vertex. Without loss of generality, o lies at the origin, edges α_T and β_T intersect the line $y=0$ at the origin with angles θ_{α_T} and θ_{β_T} respectively, and $\theta_{\beta_T} > \theta_{\alpha_T} \ge 0$. Suppose κ_T is a line segment, and q lies on κ_T.

Case 1a) Suppose κ_T is a line segment, and suppose q lies on κ_T.Without loss of generality, κ_T is supported by the line $x = r_T$. The equation of the line in polar coordinates is $r = r_T/\cos\theta$. We have $\theta_{\beta_T} \le \pi/2$. Let θ be any value such that $\theta_{\alpha_T} \le \theta \le \theta_{\beta_T}$. Suppose we choose λ such that λ's endpoint on κ_T lies at angle θ relative to the x-axis. As mentioned, the optimal choice for θ guarantees $\mathrm{cl}(\lambda) \ge \mathrm{cl}(p)$. We say θ is *feasible* if $\mathrm{cl}(\lambda) \ge \mathrm{cl}(p)$ and $\theta_{\alpha_T} \le \theta \le \theta_{\beta_T}$ (see Figure 4a).

Recall Corollary 2.1. Restricting ourselves to feasible values of θ, we have

$$\begin{aligned}\mu(\gamma(p,\lambda)) &= \ln\frac{\mathrm{cl}\,(\lambda)}{\mathrm{cl}(p)} + \theta_{\beta_T} - \theta \\ &= \ln\frac{r_T/\cos\theta}{\mathrm{cl}(p)} + \theta_{\beta_T} - \theta.\end{aligned}$$

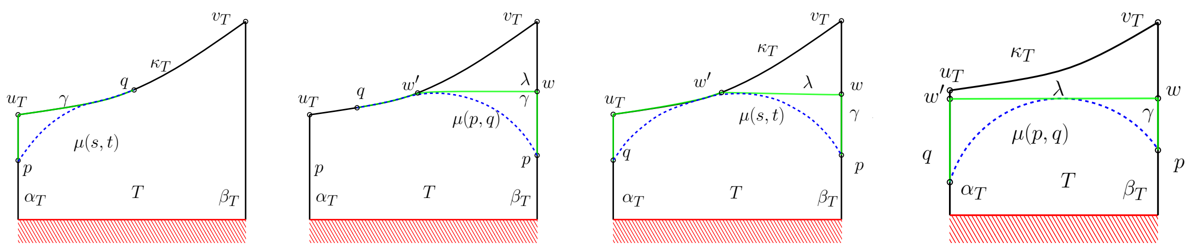
Taking the derivative, we see

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mu(\gamma(p,\lambda)) = \tan\theta - 1.$$

This expression is negative for $\theta = 0$, positive near $\theta = \pi/2$, and it has at most one root within feasible values of θ, namely at $\theta = \pi/4$. Therefore, $\mu(\gamma(p,\lambda))$ is minimized when either $\mathrm{cl}(\lambda) = \mathrm{cl}(p)$ or $\theta = \theta^* = \min\{\max\{\pi/4, \theta_{\alpha_T}\}, \theta_{\beta_T}\}$. We pick w^*_κ so that $\mathrm{cl}(w^*_\kappa) = r_T/\cos(\theta^*)$.

Case 1b) Suppose κ_T is a parabolic arc, and suppose q lies on κ_T. Without loss of generality, the parabola supporting κ_T is equidistant between o and the line $x = 2r_T$. The equation of the parabola in polar coordinates is $r = 2r_T/(1+\cos\theta)$. We have $\theta_{\beta_T} \le \pi$. Define θ and choose λ as in Case 1a. Again, angle θ is feasible if $\mathrm{cl}(\lambda) \ge \mathrm{cl}(p)$ and $\theta_{\alpha_T} \le \theta \le \theta_{\beta_T}$.

Recall Corollary 2.1. Restricting ourselves to



(a) Case 1 (b) Case 2 (c) Case 3, $w' \in \kappa_T$ (d) Case 3, $w' \in \alpha_T$

Figure 3: Different cases considered in the proof of Lemma 3.1 for a line-segment obstacle.

feasible values of θ, we have

$$\mu(\gamma(p,\lambda)) = \ln \frac{\mathrm{cl}(\lambda)}{\mathrm{cl}(p)} + \theta_{\beta_T} - \theta = \ln \frac{2r_T/(1+\cos\theta)}{\mathrm{cl}(p)} + \theta_{\beta_T} - \theta.$$

Here,

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\mu(\gamma(p,\lambda)) = \frac{\sin\theta}{1+\cos\theta} - 1 = \tan(\theta/2) - 1.$$

Again, the expression is negative for $\theta = 0$, positive for θ near π, and it has at most one root within feasible values of θ, namely at $\theta = \pi/2$. Therefore, $\mu(\gamma(p,\lambda))$ is minimized when either $\mathrm{cl}(\lambda) = \mathrm{cl}(p)$ or $\theta = \theta^* = \min\{\max\{\pi/2, \theta_{\alpha_T}\}, \theta_{\beta_T}\}$. We pick w^*_κ so that $\mathrm{cl}(w^*_\kappa) = 2r_T/(1+\cos(\theta^*))$.

Case 1c) Suppose q lies on α_T. By Corollary 2.1, the cost of λ is simply $\theta_{\beta_T} - \theta_{\alpha_T}$. Therefore, $\mu(\gamma(p,\lambda))$ is minimized when $\mathrm{cl}(\lambda) = \mathrm{cl}(p)$. We (arbitrarily) pick w^*_α so that $\mathrm{cl}(w^*_\alpha) = \mathrm{cl}(u_T)$.

Case 2) Suppose o is a polygon edge. Without loss of generality, o lies on the line $y = 0$, the edge α_T lies on the line $x = x_{\alpha_T}$, the edge β_T lies on the line $x = x_{\beta_T}$, and $x_{\beta_T} > x_{\alpha_T} \geq 0$.

Case 2a) Suppose κ_T is a line segment, and suppose q lies on κ_T. Without loss of generality, the line supporting κ_T intersects o at the origin with angle θ_κ. Let x be any value such that $x_{\alpha_T} \leq x \leq x_{\beta_T}$. Suppose we choose λ such that λ's endpoint on κ_T has x-coordinate x. As mentioned, the optimal choice for x guarantees $\mathrm{cl}(\lambda) \geq \mathrm{cl}(p)$. We say x is feasible if $\mathrm{cl}(\lambda) \geq \mathrm{cl}(p)$ and $x_{\alpha_T} \leq x \leq x_{\beta_T}$ (see Figure 4b).

Recall Corollary 2.1. Restricting ourselves to feasible values of x, we have

$$\mu(\gamma(p,\lambda)) = \ln \frac{\mathrm{cl}(\lambda)}{\mathrm{cl}(p)} + \frac{\|\lambda\|}{\mathrm{cl}\,(\lambda)} = \ln \frac{x\tan\theta_\kappa}{\mathrm{cl}(p)} + \frac{x_{\beta_T} - x}{x\tan\theta_\kappa}.$$

We see

$$\frac{\mathrm{d}}{\mathrm{d}x}\mu(\gamma(p,\lambda)) = \frac{1}{x} - \frac{x_{\beta_T}}{x^2\tan\theta_\kappa} = \frac{x\tan\theta_\kappa - x_{\beta_T}}{x^2\tan\theta_\kappa}.$$

This expression is negative for x near 0, positive for large x, and it has at most one root within feasible values of x, namely at $x = x_{\beta_T}/\tan\theta_\kappa$. Therefore, $\mu(\gamma(p,\lambda))$ is minimized when either $\mathrm{cl}(\lambda) = \mathrm{cl}(p)$ or $x = x^* = \min\{\max\{x_{\beta_T}/\tan\theta_\kappa, x_{\alpha_T}\}, x_{\beta_T}\}$. We pick w^*_κ so that $\mathrm{cl}(w^*_\kappa) = x^*\tan\theta_\kappa$.

Case 2b) Suppose κ_T is a parabolic arc, and suppose q lies on κ_T. Without loss of generality, the parabola supporting κ_T is equidistant between o and a point located at $(0, 2y_\kappa)$. Therefore, the parabola is described by the equation $y = x^2/(4y_\kappa) + y_\kappa$. Define x and choose λ as in Case 2a. Again, we say x is feasible if $\mathrm{cl}(\lambda) \geq \mathrm{cl}(p)$ and $x_{\alpha_T} \leq x \leq x_{\beta_T}$.

Recall Corollary 2.1. Restricting ourselves to feasible values of x, we have

$$\mu(\gamma(p,\lambda)) = \ln \frac{\mathrm{cl}(\lambda)}{\mathrm{cl}(p)} + \frac{\|\lambda\|}{\mathrm{cl}\,(\lambda)} = \ln \frac{x^2/(4y_\kappa) + y_\kappa}{cl(p)} + \frac{x_{\beta_T} - x}{x^2/(4y_\kappa) + y_\kappa}.$$

We have

$$\frac{\mathrm{d}}{\mathrm{d}x}\mu(\gamma(p,\lambda)) = \frac{2x^3 + 4y_\kappa x^2 + 8y_\kappa(y_\kappa - x_{\beta_T})x - 16y_\kappa^3}{(x^2 + 4y_\kappa^2)^2}.$$

This expression is negative for x near 0 and positive for large x. The derivative of the numerator is $6x^2 + 8y_\kappa x + 8y_\kappa(y_\kappa - x_{\beta_T})$, which has at most one positive root. Therefore, the numerator has at most one positive local maximum or minimum. We see $\frac{\mathrm{d}}{\mathrm{d}x}\mu(\gamma(p,\lambda))$ goes from negative to positive around exactly one positive root (which may not be feasible), and $\mu(\gamma(p,\lambda))$ has one minimum at a positive value of x. Let x' be this root of $\frac{\mathrm{d}}{\mathrm{d}x}\mu(\gamma(p,\lambda))$.

Value $\mu(\gamma(p,\lambda))$ is minimized when either $\text{cl}(\lambda) = \text{cl}(p)$ or $x = x^* = \min\{\max\{x', x_{\alpha_T}\}, x_{\beta_T}\}$. We pick w^*_κ so that $\text{cl}(w^*_\kappa) = (x^*)^2/(4y_\kappa) + y_\kappa$.

Case 2c) Suppose q lies on α_T. Let y be any value such that $0 \le y \le \text{cl}(u_T)$. Suppose we choose λ such that λ has clearance y. As mentioned, the optimal choice for y guarantees $y \ge \text{cl}(p)$. We say y is feasible if $y \ge \text{cl}(p)$ and $0 < y \le \text{cl}(u_T)$.

Recall Corollary 2.1 Restricting ourselves to feasible values of y, we have

$$\begin{aligned}\mu(\gamma(p,\lambda)) &= \ln\frac{\text{cl}(\lambda)}{\text{cl}(p)} + \frac{\|\lambda\|}{\text{cl}\,(\lambda)}\\ &= \ln\frac{y}{cl(p)} + \frac{x_{\beta_T} - x_{\alpha_T}}{y}.\end{aligned}$$

We have

$$\frac{\mathrm{d}}{\mathrm{d}y}\mu(\gamma(p,\lambda)) = 1/y - \frac{(x_{\beta_T} - x_{\alpha_T})}{y^2}.$$

This expression is negative for y near 0, positive for large y, and it has at most one root within feasible values of y, namely at $y = x_{\beta_T} - x_{\alpha_T}$. If $\text{cl}(p)$ and $x_{\beta_T} - x_{\alpha_T}$ are at most $\text{cl}(u_T)$, then $\mu(\gamma(p,\lambda))$ is minimized when either $\text{cl}(\lambda) = \text{cl}(p)$ or $y = y^* = x_{\beta_T} - x_{\alpha_T}$. If either $\text{cl}(p)$ or $x_{\beta_T} - x_{\alpha_T}$ are greater than $\text{cl}(u_T)$, then it costs less for λ to intersect κ_T than for it to intersect α_T. Therefore, if $y^* \le \text{cl}(u_T)$, we pick w^*_α so that $\text{cl}(w^*_\alpha) = y^*$. Otherwise, we set $w^*_\alpha = w^*_\beta$.

4 Approximation Algorithms

In this section, we propose a near-quadratic-time $(1+\varepsilon)$-approximation algorithm for computing the minimum-cost path. We first give a high-level overview of the algorithm and then describe each step in detail. Throughout this section, let γ^* denote a minimal-cost (s,t)-path.

High-level description. Our algorithm begins by computing the refined Voronoi diagram $\tilde{\mathcal{V}}$ of $\mathcal{O}$. The algorithm then works in three stages. The first stage computes an $O(n)$-approximation of $d^* = \mu(s,t)$, i.e., it returns a value $\tilde{d}$ such that $d^* \le \tilde{d} \le cnd^*$ for some constant $c > 0$. By augmenting $\tilde{\mathcal{V}}$ with a linear number of additional edges, each a constant-clearance path between two points on the boundary of a cell of $\tilde{\mathcal{V}}$, the algorithm constructs a graph G_1 with $O(n)$ vertices and computes a minimal-cost path from s to t in G_1.

Equipped with the value $\tilde{d}$, the second stage computes an $O(1)$-approximation of d^*. For a given $d \ge 0$, this algorithm constructs a graph G_2 by sampling $O(n)$ points on the boundary of each cell T of $\tilde{\mathcal{V}}$ and connecting these sample points by adding $O(n)$ edges (besides the boundary of T), each of which is again a constant-clearance path. The resulting graph G_2 is planar, so a minimum-cost path in G_2 from s to t can be computed in $O(n^2)$ time [9]. We show that if $d \ge \tilde{d}$, then the cost of the optimal path from s to t in G_2 is $O(d)$. Therefore, if $d \in [d^*, 2d^*]$, the cost of the optimal path is $O(d^*)$. Using the value of $\tilde{d}$, we run the above procedure for $O(\log n)$ different values of d, namely $d \in \{\tilde{d}/2^i \mid 0 \le i \le \lceil \log_2 cn \rceil\}$, and return the least costly path among them. Let $\hat{d}$ be the cost of the path returned.

Finally, using the value $\hat{d}$, the third stage samples $O(n/\varepsilon)$ points on the boundary of each cell T of $\tilde{\mathcal{V}}$ and connects each point to $O((1/\varepsilon)\log n/\varepsilon)$ other points on the boundary of T by an edge. Unlike the last two stages, each edge is no longer a constant-clearance path but it is a minimal-cost path between its endpoints lying inside T. The resulting graph G_3 has $O(n^2/\varepsilon)$ vertices and $O((n^2/\varepsilon^2)\log(n/\varepsilon))$ edges. The overall algorithm returns the minimal-cost path in G_3.

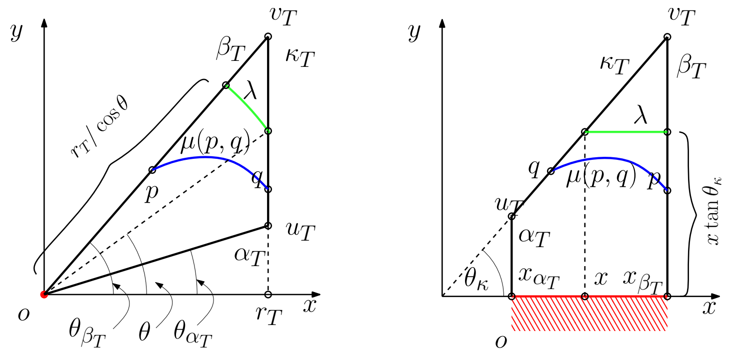
The analysis of all three stages relies on the guarantees of Lemma 3.2 concerning the existence of anchor points and well-behaved paths between points on the boundary of each cell of $\tilde{\mathcal{V}}$.

4.1 $O(n)$-approximation algorithm Here, we describe a near-linear time algorithm to obtain an $O(n)$-approximation of d^*. We augment $\tilde{\mathcal{V}}$ with $O(n)$ additional edges as described below to create the graph G_1.

We do the following for each cell T of $\tilde{\mathcal{V}}$. We compute anchor points $w^*_{\alpha_T}$ and $w^*_{\kappa_T}$ as described in Lemma 3.2. We subdivide β_T at $w^*_{\alpha_T}$ and $w^*_{\kappa_T}$, and add constant-clearance line segments or circular arcs λ_{α_T} and λ_{κ_T} from $w^*_{\alpha_T}$ and $w^*_{\kappa_T}$ respectively to the other points of equal clearance on the edges of T. Finally, let w_s be the point on β_T of clearance $\min\{\text{cl}(v_T), \text{cl}(s)\}$. We subdivide β_T at w_s and add the constant clearance path λ_s from w_s to the other point of equal clearance on T's edges. See Figure 9. All edges in G_1 are assigned the cost of their path using (1.1) and the equations of Wein *et al.* [17] for Voronoi edges. We compute and return the minimal-cost path in G_1 from s to t.

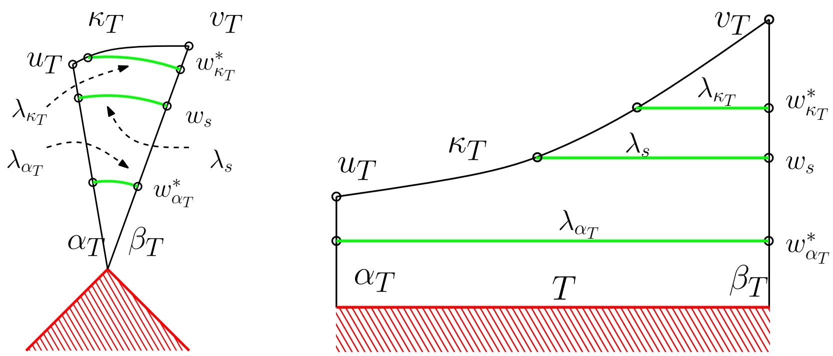
Lemma 4.1. *Graph G_1 contains an s,t-path of cost at most $O(n) \cdot d^*$.*

Proof. Suppose γ^* has points outside G_1 in the interior of Voronoi cell T. We use the notation given above for adding edges to G_1 within T. Let p and q



(a) Case 1a (b) Case 2a

Figure 4: Sample of cases considered in the proof of Lemma 3.2.



(a) Point-obstacle cell (b) Line-obstacle cell

Figure 5: Edges added within cell T for the $O(n)$-approximation algorithm.

be the first and last intersection of γ^* and T. We construct a well-behaved (p, q)-path γ through G_1 such that $\mu(\gamma) = O(d^*)$. If neither p nor q lie on β_T, then γ is simply the path guaranteed by Lemma 3.2.

Suppose otherwise, and let p lie on β_T. Suppose q lies on κ_T. Let w be point of higher clearance between $w^*_{\kappa_T}$ and w_s, and let λ be the path of higher clearance between λ_{κ_T} and λ_s. Path γ follows β_T from p to w_s, follows β_T from w_s to w, follows λ, and then follows κ_T to q. Note that γ may use some points of β_T twice; we describe it as we do to simplify the analysis.

By Corollary 2.1, $\mu(p, w_s) \leq d^*$. Therefore, $\mu(w_s, q) \leq \mu(w_s, p) + \mu(p, q) \leq 2d^*$. From Lemma 3.2, we have $\mu(\gamma[w_s, q]) \leq 7\mu(w_s, q) = O(d^*)$. Finally, $\mu(\gamma) = \mu(p, w_s) + \mu(\gamma[w_s, q]) = O(d^*)$. A similar construction is used if q lies on α_T.

We replace $\gamma^*[p, q]$ with γ, reducing the number of cells containing points in γ^* disjoint from G_1. After repeating this procedure in $O(n)$ different Voronoi cells, we create a path through G_1 with cost $O(n) \cdot d^*$.

Diagram $\tilde{\mathcal{V}}$ contains $O(n)$ vertices, edges, and cells. Graph G_1 contains a constant number of additional vertices and edges per cell, so it has $O(n)$ vertices and edges total. Computing the shortest s, t-path in G_1 takes $O(n \log n)$ time[5].

Theorem 4.2. *Let $\mathcal{O}$ be a set of polygonal obstacles in the plane, and let s, t be two points outside $\mathcal{O}$. There exists an $O(n \log n)$-time $O(n)$-approximation algorithm for computing the minimum-cost path between s and t.*

4.2 Constant-factor approximation Recall that, given an estimate d of the cost d^* of the optimal path, we construct a planar graph G_2 by sampling points along the edges of the refined Voronoi diagram $\tilde{\mathcal{V}}$. The sampling procedure here can be thought of as a warm-up for the sampling procedure given in Section 4.3.

[5]We note that G_1 is planar, so we can find shortest paths in G_1 in linear time [9]. However, using the linear time algorithm will not improve the asymptotic running time of our $O(n)$-approximation.

Let T be a Voronoi cell of $\tilde{\mathcal{V}}$ and assume without loss of generality that $\mathrm{cl}(s) \leq \mathrm{cl}(t)$. Let $w_{\min}$ and $w_{\max}$ be the points on β_T with clearance $\mathrm{cl}(t)/\exp(d)$ and $\min\{v_T, \mathrm{cl}(s)\cdot\exp(d)\}$ respectively. We place sample points on β_T between points $w_{\min}$ and $w_{\max}$ inclusive to act as vertices in G_2. The samples are chosen so the cost between consecutive samples is exactly $\frac{d}{n}$ (except possibly at one endpoint). Given a sample point p on an edge of $\tilde{\mathcal{V}}$, it is straightforward to compute the coordinates of the sample point p' on the same edge such that $\mu(p, p') = c$ for any $c > 0$. Simply use the formula for the cost along a Voronoi edge given in [17, Corollary 8]. We emphasize that the points are separated evenly by *cost*; the samples will not be uniformly placed in terms of the Euclidean distance along the arc. Figure 6 shows the samples used for our constant-factor approximation algorithm as well as our third algorithm described below.

From each sample point, we add a constant-clearance edge to G_2 within T to the other point on the edges of T with the same clearance, subdividing the edges as necessary. We also add constant clearance edges within T from anchor points w^*_α and w^*_κ as defined in Lemma 3.2. The refined Voronoi diagram $\tilde{\mathcal{V}}$ is planar. Every edge added to create G_2 stays within a single cell of $\tilde{\mathcal{V}}$ and has constant clearance. Therefore, no pair of new edges cross and G_2 is planar as well. We compute the shortest path from s to t in G_2 using the linear (in graph size) time algorithm of Henzinger *et al.* [9].

Lemma 4.3. *Suppose $d \geq d^*$ and $\mathrm{cl}(s) \leq \mathrm{cl}(t)$. Value $\mathrm{cl}(t)/\exp(d)$ is a lower bound on the minimal clearance attained by γ^* and $\mathrm{cl}(s)\cdot\exp(d)$ is an upper bound on the maximal clearance attained by γ^*.*

Proof. Let $p_{\min}$ be the point where γ^* attains the minimal clearance. Clearly, $\mu(s,t) \geq \mu(s, p_{\min}) + \mu(p_{\min}, t)$. Using this observation together with Corollary 2.1, we obtain our lower bound. The upper bound follows by similar arguments.

For each Voronoi cell T, let $\hat{\beta_T}$ be the portion of β_T that receives sample points. We have the following two properties: (i) cost $\mu(\hat{\beta_T}) = O(d)$ and (ii) if $d \geq d^*$, then no point on $\beta_T \setminus \hat{\beta_T}$ can lie on γ^*. Property (i) follows by (2.4). Property (ii) follows from Lemma 4.3.

Lemma 4.4. *Suppose $d \geq d^*$. Graph G_2 contains an s,t-path of cost at most $O(d)$.*

Proof. Let γ be a maximal portion of γ^* lying in a single Voronoi cell T of $\tilde{\mathcal{V}}$, and let p and q be the endpoints of γ. If neither p nor q lie on β_T, then Lemma 3.2 guarantees the edges of T contain a well behaved path γ' of cost at most $7\mu(\gamma)$.

Suppose otherwise, and let p lie on β_T without loss of generality. By property (ii), there exists a sample point p' on β_T such that $\mu(p,p') \leq \frac{d}{n}$. We have $\mu(p', q) \leq \mu(p', p) + \mu(p, q) \leq \mu(p,q) + \frac{d}{n}$. By Lemma 3.2 and the choice of edges in G_2, there exists a well-behaved path in G_2 through T of cost at most $7\mu(p', q)$; if this path enters the interior of T, then it does so at one of p', w^*_α, or w^*_β. In particular, there exists a path γ' in G_2 from p to q of cost at most $7\mu(p,q) + O(\frac{d}{n})$.

Each edge of $\tilde{\mathcal{V}}$ is a minimal-cost path. Therefore, each edge is incident to at most two maximal subpaths of γ^* internally disjoint from $\tilde{\mathcal{V}}$. We conclude there are $O(n)$ such subpaths. Each can be replaced by one going through G_2 as described above. The total cost of the new path from s to t is $7d^* + O(n)\cdot O(\frac{d}{n}) = O(d)$.

Property (i) ensures that the number of vertices added along each edge e is $O(n)$. Therefore, the total number of vertices added along edges of $\tilde{\mathcal{V}}$ is $O(n^2)$. We use a linear-time algorithm [9] to compute a shortest path in planar graph G_2, so constructing G_2 and finding a minimal-cost path G_2 takes $O(n^2)$ time.

For our constant-factor approximation algorithm, we perform an exponential search over the values of path costs. Let $\tilde{d} \leq cnd^*$ be the cost of the path returned by the $O(n)$-approximation algorithm (Section 4.1). For each i from 0 to $\lceil \log cn \rceil$, we take $d = \tilde{d}/2^i$ as the estimate of d^*, and run the above procedure to construct a graph G_2 and compute a minimal-cost path in the graph. We return the least costly of the paths computed over all iterations.

Fix integer $\hat{i}$ so $d^* \leq \tilde{d}/2^{\hat{i}} \leq 2d^*$. Let d_i be the cost of the shortest path in G_2 during iteration i. Let $\hat{d}$ be the minimal output of the $O(1)$-approximation algorithm over the set of $O(\log n)$ iterations. By Lemma 4.4, we have

$$\hat{d} \leq d_{\hat{i}} \leq O(\tilde{d}/2^{\hat{i}}) = O(d^*).$$

Theorem 4.5. *Let $\mathcal{O}$ be a set of polygonal obstacles in the plane, and let s, t be two points outside $\mathcal{O}$. There exists an $O(n^2 \log n)$ time $O(1)$-approximation algorithm for computing the minimum cost path between s and t.*

4.3 Computing the final approximation Finally, let $\hat{d}$ be the estimate returned by our constant factor approximation algorithm so that $d^* \leq \hat{d} \leq cd^*$ for some constant c. We construct a graph G_3 by sampling points along the edges of the refined Voronoi diagram $\tilde{\mathcal{V}}$.

Sample vertices in G_3. Let T be a Voronoi cell[6] of $\tilde{\mathcal{V}}$ and assume without loss of generality that $\text{cl}(s) \leq \text{cl}(t)$. In each case below, points within a single region are sampled so they lie at cost $\frac{\varepsilon \hat{d}}{n}$ apart. Along α_T, we place samples between points $w_{\min}$ and $w_{\max}$ with clearance $\text{cl}(t)/\exp(\hat{d})$ and $\min\{u_T, \text{cl}(s) \cdot \exp(\hat{d})\}$, respectively (including at $w_{\min}$ and $w_{\max}$). Along β_T, we place samples between points $w_{\min}$ and $w_{\max}$ with clearance $\text{cl}(t)/\exp(\hat{d})$ and $\min\{v_T, \text{cl}(s) \cdot \exp(\hat{d})\}$, respectively. Along κ_T, we place samples between u_T and the point on κ_T of cost $2\hat{d}$ from u_T. Additionally, let v' be the point of clearance $\min\{\text{cl}(s) \cdot \exp(\hat{d}), \text{cl}(v_T)\}$ on κ_T. We place samples on κ_T between v' and the point u' on κ_T of cost $4\hat{d}$ from v' such that $\text{cl}(u') \leq \text{cl}(v')$. See Figure 6.

The edges of G_3. Let T be a cell of $\tilde{\mathcal{V}}$ incident to obstacle feature o. We say two points p and q in T are *visible* to one another if the minimal-cost path from p to q relative only to o lies within T. Equivalently, the minimal-cost path relative to o is equal to the minimal-cost path relative to $\mathcal{O}$.

Let p be a sample point on β_T. We compute anchor point w^*_κ as described in Lemma 3.2. We then compute a collection of sample points $S(p)$ on κ_T as candidate neighbors of p in G_3. Let $\hat{\kappa_T}$ denote the portion of κ_T that receives sample points. Let η denote one of the two connected regions of equally spaced sample points on κ_T as described above. We add the following points to $S(p)$. For each $w \in \{p, w^*_\kappa\}$ (lying on β_T), let $\downarrow(w)$ be the sample point on κ_T of highest clearance less than $\text{cl}(w)$ in η (assuming such a point exists). Let $\uparrow(w)$ be the sample point on κ_T of lowest clearance greater than $\text{cl}(w)$ in η.

Let $i_{\max}$ be the greatest i such that $(1+\varepsilon)^i$ is at most the number of sample points on η.

To begin, we add to $S(p)$ the endpoints of η. For each $q_0 \in \{\downarrow(p), \downarrow(w^*_\kappa)\}$, we add the following points to $S(p)$. First, we add q_0 to $S(p)$. We then iteratively walk along each sample point of η in decreasing order of clearance starting with q_0. For each non-negative integer $i \leq i_{\max}$, we add the point q_i encountered at step $\lfloor (1+\varepsilon)^i \rfloor$ of the walk. Similarly, for each $q_0 \in \{\uparrow(p), \uparrow(w^*_\kappa)\}$, we add to $S(p)$ the point q_0 and perform the walk along points of *greater* clearance. See Figure 7.

We add edges from p to each visible point q of $S(p)$. The cost of the edge (p, q) is the cost of the optimal path from p to q with respect to the feature o, as given in (2.2) or (2.3).

[6]Note that as we consider each cell independently, we actually consider each edge twice. However, this does not change the complexity of the algorithm or its analysis and simplifies the description.

We add a similar set of points from each η of κ_T or α_T to $S(p)$ as well as edges to those visible members of $S(p)$. Finally, a similar procedure exists for sample points p on α_T. For such p, set $S(p)$ contains points on β_T and edges are added to visible members of that set. We compute the minimal-cost path from s to t in G_3 using Dijkstra's algorithm with Fibonacci heaps [8].

For each edge e of $\tilde{\mathcal{V}}$, we will show the same two properties as in Section 4.2 (the properties are amended naturally to use $\hat{d}$ instead of d). By our choice of sampling intervals, property (i) holds. In particular, $i_{\max} = O(\frac{1}{\varepsilon} \log \frac{n}{\varepsilon})$ when computing each set $S(p)$. We now show property (ii) holds. Let γ be any s,t-path that passes through T such that $\mu(\gamma) \leq \hat{d}$ (assuming one exists). Let p and q be the points where γ enters and exits T, respectively. We will show that q lies within the regions sampled.

If q lies on an internal arc then $\text{cl}(t)/\exp(\hat{d}) \leq \text{cl}(q) \leq \text{cl}(s) \cdot \exp(\hat{d})$. The claim holds. If q lies on the external arc κ_T and p lies on α_T, then by the analysis of Case 1 in Lemma 3.1, $\mu(u_T, q) \leq 2\mu(\gamma_T) \leq 2\hat{d}$. Point q lies in a sampled region. If not, then q lies on κ_T and p lies on β_T. Following the lines of the analysis of Case 2 in Lemma 3.1, $\mu(v', q) \leq 4\mu(\gamma_T) \leq 4\hat{d}$. Again, q lies in a sampled region.

We have the following lemmas.

Lemma 4.6. *Let T be a cell of $\tilde{\mathcal{V}}$ incident to obstacle feature o. Let p be a point on T's edges, and let e_0 be an edge of T not containing p. Let Q_p be the set of points on e_0 visible to p. If Q_p is non-empty, then it is connected and has one boundary at an endpoint of e_0.*

Proof. We consider two main cases.

Case 1) Suppose o is a polygon vertex. Without loss of generality, o lies at the origin, edge α_T intersects the line $y = 0$ at the origin with angle θ_{α_T}, edge β_T intersects the line $y = 0$ at the origin with angle θ_{β_T}, and $\theta_{\beta_T} > \theta_{\alpha_T} \geq 0$. Let $q \in Q_p$. If q does not exist, then there is nothing to prove. Otherwise, let γ be the minimal-cost path from p to q. Equation (2.2) describes the cost of γ. We consider a mapping $f : \mathbb{R} \times S^1 \rightarrow S^1 \times \mathbb{R}$ taking points in polar coordinates to a transformed plane. Given a point (r, θ), the mapping f is defined as $f(r, \theta) = (\theta, \ln r)$. Given a path γ', we abuse notation and let $f(\gamma') = f \circ \gamma'$, the composition of f and γ'. Path γ becomes a straight-line segment connecting p and q in the transformed plane. Both α_T and β_T become vertical rays in the transformed plane going

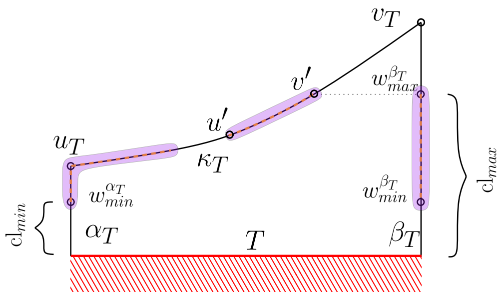


Figure 6: Samples placed on the edges of a cell T of $\tilde{\mathcal{V}}$. The sampled regions are depicted in purple. For the constant-factor approximation algorithm, samples are placed on β_T only.

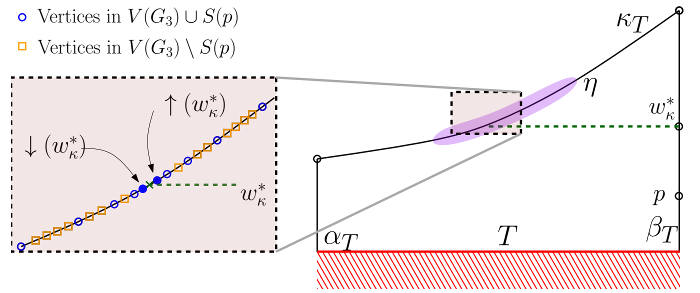
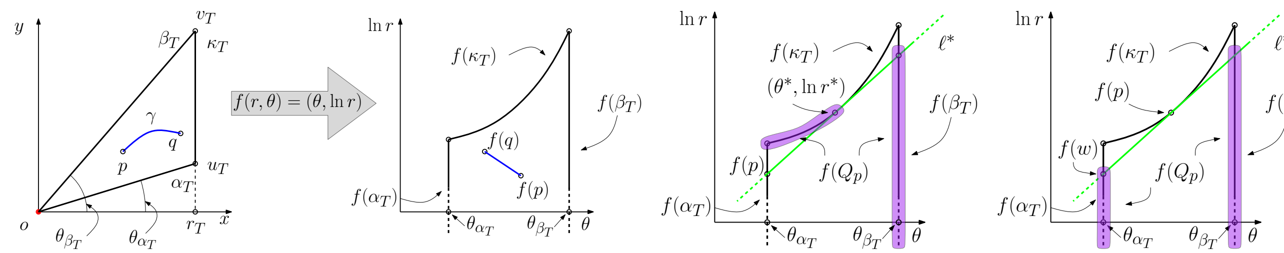


Figure 7: Vertices in $S(p)$ which are used to construct the set of edges of G_3.



(a) Transformed plane for a point-obstacle cell

(b) $p \in \alpha_T$

(c) $p \in \kappa_T$

Figure 8: Case 1 of proof of Lemma 4.6. (a) A cell of a point obstacle in $\tilde{\mathcal{V}}$ and the optimal path between points p and q (blue) in the original (left) and transformed plane (right), respectively. (b,c) The set of visible points Q_p (purple) to the point p in the transformed plane.

to $-\infty$. Further, it is straightforward to show that κ_T becomes a convex curve in the transformed plane when restricted to values of θ such that $\theta_{\alpha_T} \leq \theta \leq \theta_{\beta_T}$. Two points p' and q' in T are visible if and only if the line segment between $f(p')$ and $f(q')$ does not cross any component of T in the transformed plane (see Figure 8a).

Suppose p lies on α_T (Figure 8b). Point u_T is clearly visible to p. Let $\mathcal{L}_p$ be the set of lines in the transformed plane that intersect the point $f(p)$. The transformed line segment connecting $f(p)$ to $f(u_T)$ is supported by the (vertical) line $\ell_\infty \in \mathcal{L}_p$ with infinite slope. Consider ordering the lines in $\ell \in \mathcal{L}_p$ by *decreasing slope* starting with ℓ_∞. The first intersection of each $\ell \in \mathcal{L}_p$ with $f(\kappa_T)$ moves farther to the right as the slope decreases. Moreover, each of these first intersections are visible to p. This is true until some line ℓ^* goes tangent to $f(\kappa_T)$ at $(\theta^*, \ln r^*)$, and no line intersects $f(\kappa_T)$ again after that point. Line ℓ^* is the first line to intersect β_T without first crossing $f(\kappa_T)$. If $e_0 = \kappa_T$, then Q_p consists of all points q such that $f(q)$ lies on $f(\kappa_T)$ between $f(u_T)$ and $(\theta^*, \ln r^*)$. If $e_0 = \beta_T$, then Q_p consists of all points q such that $f(q)$ lies on β_T below the intersection of $f(\beta_T)$ and ℓ^*. A similar argument holds if p lies on β_T.

Finally, suppose p lies on κ_T and $e_0 = \alpha_T$ (Figure 8c). Let ℓ^* be the line tangent to $f(\kappa_T)$ at $f(p)$. Line ℓ^* intersects $f(\alpha_T)$ at $f(w)$. Lines to higher points on $f(\alpha_T)$ that intersect $f(p)$ cross $f(\kappa_T)$, so w is the highest clearance point on α_T visible to p. All lower points on $f(\alpha_T)$ are visible, though. Therefore, Q_p consists of all points q on α_T of clearance at most $\mathrm{cl}(w)$. A similar argument holds if p lies on κ_T and $e_0 = \beta_T$.

Case 2) Suppose o is a polygon edge. Without loss of generality, o lies on the line $y = 0$, the edge α_T lies on the line $x = x_{\alpha_T}$, the edge β_T lies on the line $x = x_{\beta_T}$, and $x_{\beta_T} > x_{\alpha_T} \geq 0$. There is no notion of the transformed plane for polygon edge o, but we are still able to use similar arguments to those given in Case 1. Two points p' and q' in T are visible if the circular arc containing p' and q' centered on o does not cross κ_T.

Suppose p lies on α_T (see Figure 9a). Point u_T is clearly visible to p. Let $\mathcal{C}_p$ be the set of circles that intersect the point p and have their center on o. The

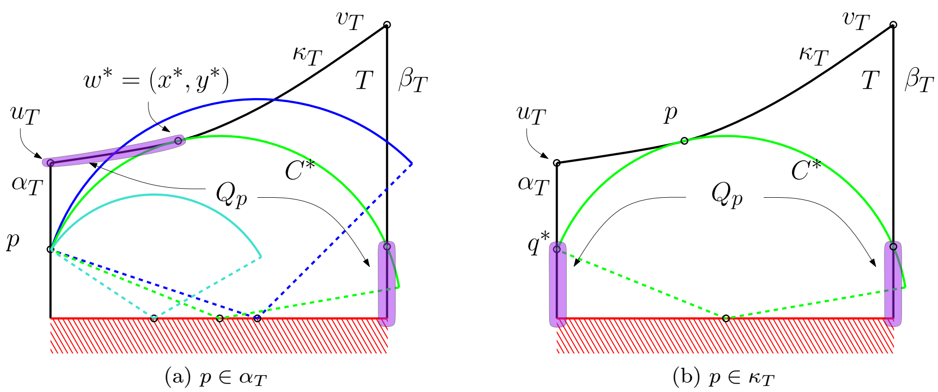


Figure 9: Case 2 of proof of Lemma 4.6. Green circle C^* which intersects p centered at $(x, 0)$ and tangent to κ_T defines the points Q_T (purple) visible to p. (a) Circles intersecting p centred at $(x, 0)$ with larger (blue) and smaller (cyan) radii than C^*, respectively.

line segment connecting p to u_T is supported by the degenerate circle $C_\infty \in \mathcal{C}_p$ with center at $(\infty, 0)$. The upper semi-circle of C_∞ contains the point (x, ∞) for every $x > x_{\alpha_T}$. Consider ordering the circles of $\mathcal{C}_p$ by decreasing x-coordinate of their centers, starting with C_∞. The upper semi-circle of each $C \in \mathcal{C}_p$ is a concave curve, and edge κ_T is a convex curve; the upper semi-circles all intersect κ_T at most two times. Fix an $x > x_{\alpha_T}$. The point (x, y) on the upper semi-circle of each $C \in \mathcal{C}_p$ moves downward as the circle centers move left until one of the circles intersects $(x, 0)$. Indeed, the bisector of line segment $p(x, y)$ must continue intersecting the center of each $C \in \mathcal{C}_p$ as the centers move left. Therefore, there is some last $C^* \in \mathcal{C}_p$ that intersects κ_T; circle C^* and κ_T are tangent at point $w^* = (x^*, y^*)$.

The first circles $C \in \mathcal{C}_p$ in our ordering lie above κ_T at each x-coordinate, but eventually they lie below κ_T at each x-coordinate. Also, the second crossing of any $C \in \mathcal{C}_p$ and κ_T can never occurs to the left of w^*. We conclude that each point (x, y) on κ_T between u_T and w^* must be visible from p. If $e_0 = \kappa_T$, then these points are precisely Q_p.

Now, assume some point on β_T is visible to p. Circle C^* is the first to reach β_T without crossing κ_T. As the center of each $C \in \mathcal{C}_p$ moves left from C^*'s center, the intersection of C and β_T moves downward. If $e_0 = \beta_T$ and Q_p is non-empty, we have Q_p consisting of all points on β_T between o and the intersection of C^* and β_T. A similar set of arguments hold if p lies on β_T.

Finally, suppose p lies on κ_T and $e_0 = \alpha_T$ (see Figure 9b). Let C^* be the circle centered on o which lies tangent to κ_T at p. Circle C^* contains a point q^* on α_T visible to p. If we initially take $C = C^*$ and move the center of C to the left, then the arc between p and α_T along C moves above κ_T and must cross again closer to α_T. However, if we move the center of C to the right, then the arc between p and α_T stays below κ_T. In addition, the intersection of C and α_T moves downward. Therefore Q_p consists of all points q such that q lies on α below q^*. A similar argument holds if p lies on κ_T and $e_0 = \beta_T$.

Lemma 4.7. *Graph G_3 contains an s,t-path of cost at most $(1 + O(\varepsilon))d^*$.*

Proof. Let γ be a maximal subpath of γ^* internally disjoint from G_3, and let p and q be the endpoints of γ. Path γ lies in Voronoi cell T. Let e_p be the edge of T containing p and e_q be the edge of T containing q. By Corollary 2.1, $e_p \neq e_q$. We assume p lies on β_T and q lies on κ_T. The other cases are the same. By Lemma 4.6 and property (ii) given above, there exists a sample point p' visible to q on β_T such that $\mu(p, p') \leq \frac{\varepsilon \hat{d}}{n}$. We have $\mu(p', q) \leq \mu(p, q) + \frac{\varepsilon \hat{d}}{n}$. Suppose there exists a point $q' \in S(p')$ on κ_T visible to p' such that $\mu(q, q') \leq \frac{\varepsilon \hat{d}}{n}$. In this case, there exists a path from p to q through G_3 which takes edge $p'q'$ and has total cost at most $\mu(p, q) + \frac{4\varepsilon \hat{d}}{n}$.

Suppose there is no visible q' as described above. Lemma 3.2 describes how to find a well-behaved path γ' between p' and q such that $\mu(\gamma') \leq 7\mu(p', q)$.

There exists $\lambda = \lambda(\gamma')$ with one endpoint on κ_T. Let w' be the endpoint of λ on κ_T. Path γ' follows κ_T from w' to q. Recall our algorithm adds sample points along several regions of length $O(\hat{d})$ such that each pair of points lies at cost $\frac{\varepsilon \hat{d}}{n}$ apart. Point q lies in one of these regions η. By assumption, q is at least $\frac{\varepsilon \hat{d}}{n}$ cost away from any sample point of η. Therefore, w' and q cannot both lie between a pair of consecutive sample points on η. Let q_0 be the first sample point of η encountered by γ' on κ_T. Path λ has an endpoint on κ_T of clearance $\mathrm{cl}(p')$ or $\mathrm{cl}(w^*_\kappa)$. Therefore, $q_0 \in \{\downarrow(p'), \downarrow(w^*_\kappa), \uparrow(p'), \uparrow(w^*_\kappa)\}$.

For each of these possible q_0, our algorithm adds samples q_i to $S(p)$ spaced geometrically away from q_0 in the direction of q. These samples include one endpoint of η. Let q_k be the last of these sample points closer to q_0 than q, and let q_{k+1} be the next of these sample points. By Lemma 4.6, at least one of q_k and q_{k+1} is visible to p. Let q' be this visible point.

Let $\delta = \mu(q_0, q)\frac{n}{\varepsilon\hat{d}}$. Value δ is an upper bound on the number of samples in η between q_0 and q. We have $\lfloor(1+\varepsilon)^k\rfloor \le \delta \le \lfloor(1+\varepsilon)^{k+1}\rfloor$. In particular $\delta \le (1+\varepsilon)^{k+1}$, which implies $\delta - \lfloor(1+\varepsilon)^k\rfloor \le \varepsilon\delta + 1$. Similarly, $\lfloor(1+\varepsilon)^{k+1}\rfloor - \delta \le \varepsilon\delta$. Also, $\mu(q_0, q) \le 7\mu(p', q)$. We have

$$\begin{aligned}\mu(q, q') &\le (\varepsilon\delta + 1)\frac{\varepsilon\hat{d}}{n} \\ &\le \left(\mu(q_0, q)\frac{\varepsilon n}{\varepsilon\hat{d}} + 1\right)\frac{\varepsilon\hat{d}}{n} \\ &= \varepsilon\mu(q_0, q) + \frac{\varepsilon\hat{d}}{n} \\ &\le 7\varepsilon\mu(p', q) + \frac{\varepsilon\hat{d}}{n}.\end{aligned}$$

We have $\mu(p', q') \le \mu(p', q) + \mu(q, q') \le (1 + 7\varepsilon)\cdot\mu(p', q) + \frac{\varepsilon\hat{d}}{n}$. There exists a path from p to q through G_3 which takes edge $p'q'$ and has total cost $(1+7\varepsilon)\mu(p, q) + O(\frac{\varepsilon\hat{d}}{n})$.

Each edge of $\tilde{\mathcal{V}}$ is a minimal-cost path. Therefore, each edge is incident to at most two maximal subpaths of γ^* internally disjoint from $\tilde{\mathcal{V}}$. We conclude there are $O(n)$ such subpaths. Each can be replaced by one going through G_3 as described above. The total cost of the new path from s to t is

$$\begin{aligned}(1+7\varepsilon)\cdot d^* + O(n)\cdot O(\frac{\varepsilon\hat{d}}{n}) &= \\ (1+O(\varepsilon))\cdot d^* + O(\varepsilon)\cdot\hat{d} &= \\ (1+O(\varepsilon))d^*.\end{aligned}$$

Recall that n denotes the number of obstacle features. The refined Voronoi diagram $\tilde{\mathcal{V}}$ contains a linear number of vertices and edges. Vertices are sampled at intervals of cost $\frac{\varepsilon\hat{d}}{n}$. Therefore, property (i) ensures that the number of vertices added along each edge e is $O(\frac{n}{\varepsilon})$. In turn, the total number of vertices in G_3 is $O(\frac{n^2}{\varepsilon})$. Each sample vertex on an internal edge is incident to $O(\frac{1}{\varepsilon}\log\frac{n}{\varepsilon})$ edges of G_3, bringing the total number of edges in G_3 to $O(\frac{n^2}{\varepsilon^2}\log\frac{n}{\varepsilon})$. Recall, we compute the shortest path from s to t using Dijkstra's algorithm with Fibonacci heaps [8].

Thus, we obtain the following theorem.

THEOREM 4.8. *Let $\mathcal{O}$ be a set of polygonal obstacles in the plane with n vertices total, and let s, t be two points outside $\mathcal{O}$. Given a parameter $\varepsilon \in (0, 1]$, there exists an $O(\frac{n^2}{\varepsilon^2}\log\frac{n}{\varepsilon})$-time approximation algorithm for the minimal-cost path problem between s and t such that the algorithm returns an s,t-path of cost $(1+\varepsilon)d^*$.*

5 Discussion and Future Work

In this paper we present the first polynomial-time approximation algorithm for the problem of computing minimal-cost paths between two given points (when using the cost defined in (1.1)). Our immediate goal is to improve the running time of our algorithm to be near-linear. A possible approach would be to refine the notion of anchor points so it suffices to put only $O(\log n)$ additional points on each edge of the refined Voronoi diagram. Finally, there are natural interesting open problems that we believe should be addressed. The first is to prove whether the problem at hand is NP-Hard or not. The second calls for extending our algorithm to compute near-optimal paths amid polyhedral obstacles in $\mathbb{R}^3$.

References

[1] Agarwal, P.K., Wang, H.: Approximation algorithms for curvature-constrained shortest paths. SIAM J. Comput. **30**(6), 1739–1772 (2000)

[2] Aurenhammer, F., Klein, R., Lee, D.: Voronoi Diagrams and Delaunay Triangulations. World Scientific (2013)

[3] Canny, J.F., Reif, J.H.: New lower bound techniques for robot motion planning problems. In: Symposium on Foundations of Computer Science, pp. 49–60 (1987)

[4] Chen, D.Z., Daescu, O., Klenk, K.S.: On geometric path query problems. Int. J. Comput. Geometry Appl. **11**(6), 617–645 (2001)

[5] Chen, D.Z., Wang, H.: Computing shortest paths among curved obstacles in the plane. In: Symposuim on Computational Geometry, pp. 369–378 (2013)

[6] Choset, H., Lynch, K.M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L.E., Thrun, S.: Principles of Robot Motion: Theory, Algorithms, and Implementation. MIT Press (2005)

[7] Cohen, M.B., Fasy, B.T., Miller, G.L., Nayyeri, A., Sheehy, D.R., Velingker, A.: Approximating nearest neighbor distances. arXiv preprint arXiv:1502.08048 (2015)

[8] Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. J. ACM **34**(3), 596–615 (1987)

[9] Henzinger, M.R., Klein, P.N., Rao, S., Subramanian, S.: Faster shortest-path algorithms for planar graphs. J. Comput. Syst. Sci. **55**(1), 3–23 (1997)

[10] Hershberger, J., Suri, S., Yıldız, H.: A near-optimal algorithm for shortest paths among curved obstacles in the plane. In: Symposium on Computational Geometry, pp. 359–368 (2013)

[11] Lebeck, N., Mølhave, T., Agarwal, P.K.: Computing highly occluded paths on a terrain. In: International Conference on Advances in Geographic Information Systems, pp. 14–23 (2013)

[12] Mitchell, J.S.B.: Shortest paths and networks. In: Handbook of Discrete and Computational Geometry, Second Edition., pp. 607–641 (2004)

[13] Mitchell, J.S.B., Rote, G., Woeginger, G.J.: Minimum-link paths among obstacles in the plan. Algorithmica **8**(5&6), 431–459 (1992)

[14] Ó'Dúnlaing, C., Yap, C.: A "retraction" method for planning the motion of a disc. J. Algorithms **6**(1), 104–111 (1985)

[15] Song, G., Miller, S., Amato, N.M.: Customizing PRM roadmaps at query time. In: IEEE International Conference on Robotics and Automation, pp. 1500–1505 (2001)

[16] Wein, R., van den Berg, J.P., Halperin, D.: The visibility-voronoi complex and its applications. Comput. Geom. **36**(1), 66–87 (2007)

[17] Wein, R., van den Berg, J.P., Halperin, D.: Planning high-quality paths and corridors amidst obstacles. I. J. Robotic Res. **27**(11-12), 1213–1231 (2008)