Approximate Euclidean Shortest Paths amid Convex Obstacles*

Pankaj K. Agarwal R. Sharathkumar Hai Yu

Department of Computer Science
Duke University, Durham NC 27708

July 3, 2008

Abstract

We develop algorithms and data structures for the approximate Euclidean shortest path problem amid a set $\mathcal{P}$ of k convex obstacles in $\mathbb{R}^2$ and $\mathbb{R}^3$, with a total of n faces. The running time of our algorithms is linear in n, and the size and query time of our data structure are independent of n. Our approeach is to quickly compute a small sketch $\mathcal{Q}$ of $\mathcal{P}$ whose size is independent of n and then compute approximate shortest paths with respect to $\mathcal{Q}$.

*Work on this paper is supported by NSF under grants CNS-05-40347, CFF-06-35000, and DEB-04-25465, by ARO grants W911NF-04-1-0278 and W911NF-07-1-0376, by an NIH grant 1P50-GM-08183-01, by a DOE grant OEG-P200A070505, and by a grant from the U.S.–Israel Binational Science Foundation.

1 Introduction

The Euclidean shortest-path problem is defined as follows: Given a set of pairwise-disjoint convex polyhedral obstacles in $\mathbb{R}^2$ or $\mathbb{R}^3$ and two points s and t, compute the shortest path between s and t that avoids the interior of the obstacles. This problem has received much attention in computational geometry and robotics; see, e.g., the survey paper [15] for a comprehensive review. Let n denote the total complexity of the obstacles. In two dimensions, Hershberger and Suri [14] presented an optimal $O(n \log n)$-time algorithm for computing the exact Euclidean shortest path between two points amid polygonal obstacles. In three dimensions, Canny and Reif [5] proved that the problem is NP-Hard even if the obstacles are a set of parallel triangles, and the best known exact algorithm runs in time singly exponential in n [20]. When the obstacles are convex, Sharir [22] showed that the exact shortest path can be computed in $n^{O(k)}$, where k is the number of obstacles.

The known lower-bound results have motivated researchers to develop fast algorithms for computing approximate shortest paths and for computing shortest paths in special cases. Papadimitriou [17] gave an $O(n^4(L + \log(n/\varepsilon))/\varepsilon^2)$-time algorithm for computing an ε-approximate shortest path, i.e., a path whose length is at most $(1 + \varepsilon)$ times the length of the shortest path. Here L is the number of bits of precision in the model of computation. A rigorous analysis of Papadimitriou's algorithm was later given by Choi *et al.* [8]; see also [4]. Clarkson [9] proposed a different algorithm for computing an ε-approximate shortest path; the running time of his algorithm is roughly $O(n^2 \log^{O(1)} n/\varepsilon^4)$ (the running time also depends on the geometry of obstacles).

The special case of computing a shortest path between two points along the surface of a single convex polytope has been widely studied. After an initial $O(n^3 \log n)$ algorithm by Sharir and Schorr [23], the bound was improved to $O(n^2)$ by Chen and Han [7] (see also [16]). A major open problem was whether the running time can be improved to $O(n \log n)$. Such an algorithm is recently developed by Shreiber and Sharir [21]. Hershberger and Suri [13] proposed a simple $O(n)$ algorithm to compute a 1-approximate shortest path. Later Agarwal *et al.*. [3] developed an $O(n \log(1/\varepsilon) + 1/\varepsilon^3)$ algorithm to compute an ε-short path (see also [2]). Their algorithm computes a convex polytope of size $O(1/\varepsilon^{3/2})$ that approximates the original polytope and runs a quadratic shortest-path algorithm on the simplified polytope. The aforementioned exact shortest-path algorithms [23, 7, 16] can also construct a data structure for a given polytope, a fixed point s and a parameter $0 < \varepsilon \leq 1$ so that the shortest distance to a query point can be answered quickly. Har-Peled [12] described a data structure of size $O((n/\varepsilon) \log(1/\varepsilon))$ that can compute an ε-short distance from source s to a query point in $O(\log(n/\varepsilon))$ time. His technique applies to the more genearal case of polyhedral obstacles, albeit with much worse preprocessing time (roughly $O(n^4/\varepsilon^6)$) and space complexity ($O(n^2/\varepsilon^{4+\delta})$ for any $\delta > 0$).

In this paper we study the problem of computing approximate Euclidean shortest paths amid convex obstacles. Let $\mathcal{P} = \{P_1, \ldots, P_k\}$ be a set of k pairwise-disjoint convex obstacles in $\mathbb{R}^d$ $(d = 2, 3)$. Each P_i is represented as the intersection of n_i halfspaces; set $n = \sum_{i=1}^{k} n_i$, which denotes the total complexity of $\mathcal{P}$. The *free space* of $\mathcal{P}$, denoted by $\mathcal{F}(\mathcal{P})$, is defined as the closure of $\mathbb{R}^3 \setminus \bigcup \mathcal{P}$. Given two points $s, t \in \mathcal{F}(\mathcal{P})$, let $\pi_{\mathcal{P}}(s, t)$ denote the shortest path between s and t in $\mathcal{F}(\mathcal{P})$, and let $d_{\mathcal{P}}(s, t)$ denote the length of $\pi_{\mathcal{P}}(s, t)$. Let $\varepsilon > 0$ be a fixed parameter. The *ε-approximate shortest-path problem* is to compute a path $\pi \subseteq \mathcal{F}(\mathcal{P})$ between s and t whose length is at most $(1 + \varepsilon)d_{\mathcal{P}}(s, t)$. Such a path π is called an *ε-short path*, and its length is called an *ε-short distance*. For a fixed source $s \in \mathcal{F}(\mathcal{P})$, the *approximate shortest-path query problem* is to preprocess $\mathcal{P}$ into a data structure so that for any query point $t \in \mathcal{F}(\mathcal{P})$, an ε-short distance (or an ε-short path) between s and t can be reported quickly.

We obtain algorithms for computing approximate shortest paths between two points whose running time depends linearly in n, and data structures for answering approximate shortest-path queries to a fixed source whose size is independent of n. As far as we know, our results are the first to achieve this kind of performance. More specifically, we obtain the following:

- In $\mathbb{R}^2$, for any two points $s, t \in \mathcal{F}(\mathcal{P})$ and a parameter $0 < \varepsilon \le 1$, an ε-short path between s and t can be computed in $O(n + (k/\sqrt{\varepsilon}) \log(k/\varepsilon))$ time.
- In $\mathbb{R}^3$, for any two points $s, t \in \mathcal{F}(\mathcal{P})$ and a parameter $0 < \varepsilon \le 1$, an ε-short path between s and t can be computed in $O(n + (k^4/\varepsilon^7) \log^2(k/\varepsilon) \log\log k)$ time.
- In $\mathbb{R}^3$, for a fixed source $s \in \mathcal{F}(\mathcal{P})$ and a parameter $0 < \varepsilon \le 1$, a data structure of size $O(k^3 \operatorname{poly}(\log k, 1/\varepsilon))$ can be constructed in $O(n \log k + k^7 \operatorname{poly}(\log k, 1/\varepsilon))$ so that an ε-short distance between s and a query point $t \in \mathcal{F}(\mathcal{P})$ can be reported in $O(\log^2(k/\varepsilon) \log(1/\varepsilon))$ time. An ε-short path can be reported in $O(k^2/\varepsilon^{3/2})$ time by increasing the size of the data structure to $O(k^5 \operatorname{poly}(\log k, 1/\varepsilon))$.

As can be seen, when $k \ll n$, our algorithms and data structures perform much better in terms of space and running time than previously known results.

This paper is organized as follows. In Section 2, we describe our results for approximate shortest paths in $\mathbb{R}^2$. In Section 3, we present our approximate shortest path algorithm in $\mathbb{R}^3$. In Section 4, we describe a data structure for answering approximate shortest-path queries in $\mathbb{R}^3$.

2 Approximate Shortest Paths in $\mathbb{R}^2$

For a parameter $\varepsilon > 0$, a set $\mathcal{Q} = \{Q_1, \dots Q_k\}$ of k pairwise-disjoint convex polygons is called an *ε-sketch* of $\mathcal{P}$ if

(i) $P_i \subseteq Q_i$, for $i = 1, \cdots, k$;

(ii) for any $s, t \in \mathcal{F}(\mathcal{Q})$, $d_{\mathcal{Q}}(s, t) \le (1 + \varepsilon) d_{\mathcal{P}}(s, t)$.

Since $\mathcal{F}(\mathcal{Q}) \subseteq \mathcal{F}(\mathcal{P})$, $\pi_{\mathcal{Q}}(s, t) \subseteq \mathcal{F}(\mathcal{P})$ for any $s, t \in \mathcal{F}(\mathcal{Q})$. Therefore (ii) implies that $\pi_{\mathcal{Q}}(s, t)$ is an ε-short path of s, t in $\mathcal{F}(\mathcal{P})$. Next we describe an algorithm to construct an ε-sketch $\mathcal{Q}$ of small complexity.

Set $r = \lceil \sqrt{2}\pi/\sqrt{\varepsilon} \rceil$ and let $u_j = (\cos j2\pi/r, \sin j2\pi/r)$ for $0 \le j < r$. The set $\mathcal{N} = \{u_j \mid 0 \le j < r\}$ is a uniform set of directions from $\mathbb{S}^1$. For a convex polygon P and a direction $u_j \in \mathcal{N}$, let $\ell_j(P)$ denote the line passing through the extreme point of P in direction u_j, and let $h_j(P)$ denote the halfplane bounded by $\ell_j(P)$ and containing P. Set $H_i' = \{h_j(P_i) \mid 0 \le j < r\}$. In addition, let H_i'' be the set of four halfplanes which define the orthogonal bounding-box of P_i. Set $H_i = H_i' \cup H_i''$.

We call a pair $\{P_i, P_j\}$ *vertically visible* if there is a vertical segment e connecting ∂P_i to ∂P_j such that the relative interior of e does not intersect any polygon of $\mathcal{P}$ (see Figure 1). Let ℓ_{ij} be the line that separates P_i and P_j. Let Φ be the set of vertically visible pairs. It can be shown that $|\Phi| \in O(k)$ and that it can be computed in $O(n + k \log n)$ time.

For each P_i, set $\mathcal{P}_i = \{P_j \mid \{P_i, P_j\} \in \Phi\}$. For each $P_j \in \mathcal{P}_i$, let g_j be the halfplane that contains P_i and that is bounded by a line parallel to ℓ_{ij} that supports P_i and that separates P_j from P_i. Set $G_i = \{g_j \mid P_j \in \mathcal{P}_i\}$. We define

$$Q_i = \Big(\bigcap_{g \in G_i} g\Big) \cap \Big(\bigcap_{h \in H_i} h\Big)$$

Set $\mathcal{Q} = \{Q_1, \dots, Q_k\}$. We next prove that $\mathcal{Q}$ is an ε-sketch of $\mathcal{P}$. We call a vertex $v \in \partial Q_i$, *new* if $v \notin \partial P_i$. Each edge of Q_i touches a vertex of P_i. For each new vertex v of Q_i, let l_v (resp., r_v) be the vertex of P_i lying on the edge adjacent to v in counterclockwise (resp., clockwise) direction. We denote by $\Delta(v)$ the triangle formed by l_v, v, and r_v. Using the fact that the internal angle of each new vertex in Q_i is at least $\pi - \sqrt{2\varepsilon}$, we can prove the following.

Lemma 1 *For any pair of points $p \in \overline{vl_v}$ and $q \in \overline{vr_v}$, $\|pv\| + \|vq\| \leq (1+\varepsilon)\|pq\| \leq (1+\varepsilon)d_{\mathcal{P}}(p,q)$.*

Proof: $\|pv\| + \|qv\| \leq \|pq\|/\sin(\angle pvq/2) \leq \|pq\|/\sin(\pi/2 - \sqrt{\varepsilon/2}) \leq \|pq\|/(1-\varepsilon/2) \leq (1+\varepsilon)\|pq\| \leq (1+\varepsilon)d_{\mathcal{P}}(p,q)$. □

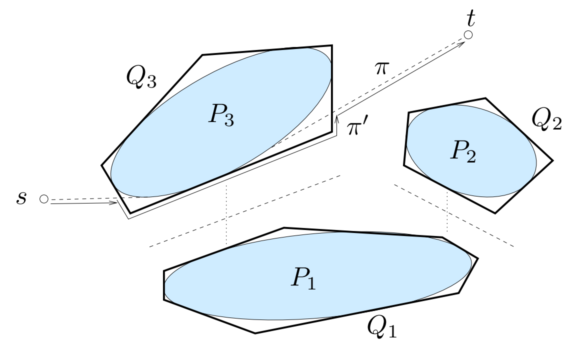

Figure 1. A path $\pi \subseteq \mathcal{F}(\mathcal{P})$ can be modified into $\pi' \subseteq \mathcal{F}(\mathcal{Q})$ so that $|\pi'| \leq (1+\varepsilon)|\pi|$.

Lemma 2 *$\mathcal{Q}$ is an ε-sketch of $\mathcal{P}$.*

Proof: By construction of H_i, the bounding box of P_i and Q_i are identical. Using this observation, it can be shown that any pair of polygons $\{Q_i, Q_j\}$ intersect if and only if there exists a vertically visible pair of polygons $\{P_i, P_k\} \in \Phi$ such that $\{Q_i, Q_k\}$ intersect. Since G_i adds halfplanes to ensure that no two vertically visible pairs intersect, we conclude that $\mathcal{Q}$ is a set of pairwise-disjoint convex polygons. Furthermore, since each halfplane in $G_i \cup H_i$ contains P_i, $P_i \subseteq Q_i$. It thus remains to prove that for any $s, t \in \mathcal{F}(\mathcal{Q})$, $d_{\mathcal{Q}}(s,t) \leq (1+\varepsilon)d_{\mathcal{P}}(s,t)$.

Set $\Sigma = \{\Delta(v) \mid v \text{ is a new vertex of some } Q_i \in \mathcal{Q}\}$. Σ consists of a set of obtuse-angled triangles whose interiors are pairwise-disjoint and which cover the region $\mathcal{F}(\mathcal{Q}) \setminus \mathcal{F}(\mathcal{P})$. For any pair of points $s, t \in \mathcal{F}(\mathcal{Q})$, let $\pi = \pi_{\mathcal{P}}(s,t)$. If π does not intersect any triangle in Σ, then $\pi \subseteq \mathcal{F}(\mathcal{Q})$ and $d_{\mathcal{Q}}(s,t) = d_{\mathcal{P}}(s,t)$. Let $\Sigma_{st} = \langle \Delta(v_1), \ldots, \Delta(v_m) \rangle \subseteq \Sigma$ be the sequence of triangles that π intersects and let $\langle (p_1, q_1), \ldots, (p_m, q_m) \rangle$ be the sequence of pairs of intersection points of π with the boundaries of triangles in Σ_{st}. Set $s = q_0$ and $t = p_{m+1}$. We obtain a path π' from π by replacing each segment $p_i q_i$ with $p_i v_i \circ v_i q_i$ (see Figure 1). Clearly, $\pi' \subseteq \mathcal{F}(\mathcal{Q})$. In addition,

$$\begin{aligned} |\pi'| &= \sum_{i=0}^{m} d_{\mathcal{Q}}(q_i, p_{i+1}) + \sum_{i=1}^{m} (\|p_i v_i\| + \|v_i q_i\|) \\ &\leq \sum_{i=0}^{m} d_{\mathcal{P}}(q_i, p_{i+1}) + (1+\varepsilon) \sum_{i=1}^{m} d_{\mathcal{P}}(p_i, q_i) \\ &\leq (1+\varepsilon)|\pi|, \end{aligned}$$

thereby implying that $d_{\mathcal{Q}}(s,t) \leq (1+\varepsilon)d_{\mathcal{P}}(s,t)$. □

Theorem 1 *Given a set $\mathcal{P}$ of k pairwise-disjoint convex polygons of total complexity n in $\mathbb{R}^2$, an ε-sketch of $\mathcal{P}$ with size $O(k/\sqrt{\varepsilon})$ can be computed in $O(n + k \log k)$ time.*

Remark. If we assume that the vertices of the input polygons in $\mathcal{P}$ are given in the sorted order, we can compute an ε-sketch $\mathcal{Q}$ of $\mathcal{P}$ in $O(k \log n)$ time.

As an immediate applications of the above theorem, we show how to compute an ε-short path between two points $s, t \in \mathcal{F}(\mathcal{P})$. We treat s and t as two additional (degenerate) obstacles and compute an ε-sketch $\mathcal{Q}$ of $\mathcal{P} \cup \{s, t\}$. This ensures that $s, t \in \mathcal{F}(\mathcal{Q})$. We then apply the the algorithm of Hershberger and Suri [14] to obtain $\pi_{\mathcal{Q}}(s, t)$; the running time is $O((k/\sqrt{\varepsilon}) \log(k/\varepsilon))$. Since $\mathcal{Q}$ is an ε-sketch, $\pi_{\mathcal{Q}}(s, t)$ is an ε-short path of s, t in $\mathcal{F}(\mathcal{P})$. Moreover, the path consists of $O(k/\sqrt{\varepsilon})$ edges.

Corollary 1 *Given a set $\mathcal{P}$ of k pairwise-disjoint convex polygons of total complexity n in $\mathbb{R}^2$ and two points $s, t \in \mathcal{F}(\mathcal{P})$, an ε-short path between s and t which consists of $O(k/\sqrt{\varepsilon})$ edges can be computed in $O(n + (k/\sqrt{\varepsilon}) \log(k/\varepsilon))$ time.*

3 Approximate Shortest Paths in $\mathbb{R}^3$

In this section we present an efficient algorithm for computing approximate shortest paths amid a set of convex polytopes in $\mathbb{R}^3$. The basic idea of our algorithm is the same as in the preceding section, i.e., to first compute a sketch of small size for the convex obstacles and then compute a path amid the sketch. However, a simple example shows that one cannot hope for a small-sized sketch that works for *all* pairs of points $s, t \in \mathcal{P}$ simultaneously. Nonetheless, we show that a small-sized sketch can indeed be computed for any *fixed* pair of points $s, t \in \mathcal{P}$, which suffices for our purpose.

Outer approximations. For a set $U \subseteq \mathbb{R}^3$ and a real number $r > 0$, let $U_r = U \oplus \mathbb{B}_r$ where $\oplus$ refers to the Minkowski sum and $\mathbb{B}_r$ denote a ball of radius r centered at the origin. For a parameter $r > 0$ and a convex polytope P of n vertices, an *outer r-approximation* of P is a convex polytope $P(r)$ such that $P \subseteq P(r) \subseteq P_r$. Set $\delta = r/\operatorname{diam}(P)$. Dudley [10] has shown that there is a polytope $P(r)$ of size $O(1/\delta)$, and Agarwal *et al.* [3] has shown that it can be computed in $O(n + (1/\delta)\log(1/\delta))$ time. The concept of outer approximation can be generalized to k pairwise-disjoint convex polytopes as follows. Given a parameter $r > 0$ and a set $\mathcal{P} = \{P_1, \ldots, P_k\}$ of k pairwise-disjoint convex polytopes in $\mathbb{R}^3$, each of diameter at most D, we call $\mathcal{Q} = \{Q_1, \cdots, Q_k\}$ an *outer r-approximation* of $\mathcal{P}$ if the convex polytopes in $\mathcal{Q}$ are pairwise-disjoint and $P_i \subseteq Q_i \subseteq (P_i)_r$ for $i = 1, \cdots, k$.

Set $\delta = r/D$. An r-outer approximation $\mathcal{Q}$ of $\mathcal{P}$ of total complexity $O(k^2 + k/\delta)$ can be computed as follows. For each $P_i \in \mathcal{P}$, we first compute Dudley's outer r-approximation $P_i(r)$ of P_i. Next, for $j \neq i$, we compute a supporting plane $h_{i,j}$ of P_i that separates P_i and P_j. Let $h_{i,j}^+$ denote the halfspace bounded by $h_{i,j}$ and containing P_i. We set $Q_i = P_i(r) \cap \bigcap_{j \neq i} h_{i,j}^+$. The resulting set $\mathcal{Q} = \{Q_1, \cdots, Q_k\}$ is a set of pairwise-disjoint convex polytopes such that $P_i \subseteq Q_i \subseteq P_i(r) \subseteq (P_i)_r$. Hence $\mathcal{Q}$ is an outer approximation of $\mathcal{P}$. Since the complexity of each $Q_i \in \mathcal{Q}$ is $O(k + 1/\delta)$, the total complexity of $\mathcal{Q}$ is $O(k^2 + k/\delta)$. Each supporting hyperplane $h_{i,j}$ can be computed by using the Dobkin-Kirkpatrick hierarchies of P_i and P_j in $O(\log |P_i| \cdot \log |P_j|) = O(\log^2 n)$ time. Hence, the time spent in computing $\mathcal{Q}$ is

$$O(n + (k/\delta)\log(1/\delta) + k^2 \log^2 n + (k^2 + k/\delta)\log(k + 1/\delta)) = O(n + k^2 \log^2 k + (k/\delta)\log(1/\delta)).$$

ε-Sketches. For two points $s, t \in \mathcal{F}(\mathcal{P})$ and a value such that $d \geq \|st\|$, we show how to construct a set $\mathcal{Q}$ of at most k pairwise-disjoint convex obstacles of total complexity $O(k^2/\varepsilon^{3/2})$ such that

$$d_{\mathcal{Q}}(s, t) \leq (1 + \varepsilon/2) d_{\mathcal{P}}(s, t) + \varepsilon d/8. \tag{1}$$

Let C_{4d} be a cube centered at s of side length $4d$. Note that t lies within C_{4d} because $d \geq \|st\|$. We clip every polytope of $\mathcal{P}$ with C_{4d} and obtain a set $\mathcal{P}'$ of at most k pairwise-disjoint convex obstacles, each of diameter at most $4\sqrt{3}d$. We treat s, t as two additional (degenerate) obstacles and compute an outer r-approximation $\mathcal{Q}$ of $\mathcal{P}' \cup \{s, t\}$ with $r = \varepsilon^{3/2} d/ck$, where c is a sufficiently large constant. Observe that s and t are in $\mathcal{F}(\mathcal{Q})$. The resulting set $\mathcal{Q}$ has total complexity $O(k^2/\varepsilon^{3/2})$ and can be constructed in time $O(n + k^2 \log^2 k + (k^2/\varepsilon^{3/2}) \log(k/\varepsilon))$. In fact, if we precompute the Dobkin-Kirkpatrick hierarhcy of each $P_i \in \mathcal{P}$ in a total of $O(n)$ time, then $\mathcal{Q}$ can be computed in an additional $O(k^2 \log^2 k + (k^2/\varepsilon^{3/2}) \log(k/\varepsilon))$ time.

Now we prove (1). We need the following lemma (whose proof is included in the appendix).

Lemma 3 *Let P and Q be two convex polytopes such that $P \subseteq Q \subseteq P_r$. For a parameter $0 < \varepsilon \leq 1$ and any pair of points $p, q \in \partial Q$,*

$$d_P(p, q) \leq d_Q(p, q) \leq (1 + \varepsilon/2) d_P(p, q) + (2\pi + 6)r + 100r/\sqrt{\varepsilon}. \tag{2}$$

If the path $\pi_{\mathcal{P}}(s, t)$ does not intersect any polytope in $\mathcal{Q}$, then $\pi_{\mathcal{Q}}(s, t) = \pi_{\mathcal{P}}(s, t)$ and therefore (1) holds. So assume that $\pi_{\mathcal{P}}(s, t)$ intersects a polytope of $\mathcal{Q}$. It is well known that or any $Q_i \in \mathcal{Q}$, the intersection $\pi_{\mathcal{P}}(s, t) \cap Q_i$ consists of at most one connected component [18]. For each polytope $Q_i \in \mathcal{Q}$ intersected by $\pi_{\mathcal{P}}(s, t)$, let $p_i, q_i \in \partial Q_i$ be the corresponding entry and exit points of $\pi_{\mathcal{P}}(s, t)$. We obtain a new path π from $\pi_{\mathcal{P}}(s, t)$ by replacing its subpath $\pi_{\mathcal{P}}(p_i, q_i)$ with $\pi_{\mathcal{Q}}(p_i, q_i)$, for each pair (p_i, q_i). Clearly, $\pi \subseteq \mathcal{F}(\mathcal{Q})$. Furthermore, for each pair (p_i, q_i), applying Lemma 3 on P_i, Q_i and $p_i, q_i \in \partial Q_i$ with $r = \varepsilon^{3/2} d/ck$, we obtain

$$d_{\mathcal{Q}}(p_i, q_i) \leq (1 + \varepsilon/2) d_{\mathcal{P}}(p_i, q_i) + \varepsilon d/8k,$$

provided c is a sufficiently large constant. Hence $|\pi| \leq (1 + \varepsilon/2) d_{\mathcal{P}}(p_i, q_i) + \varepsilon d/8$, implying (1) as desired.

Lemma 4 *Let $s, t \in \mathcal{F}(\mathcal{P})$, and let $d \geq \|st\|$ be a real value. A set $\mathcal{Q}$ of at most k pairwise-disjoint convex polytopes can be computed in $O(k^2 \log^2 k + (k^2/\varepsilon^{3/2}) \log(k/\varepsilon))$ time using the Dobkin-Kirkpatrick hierarchies of the polytopes in $\mathcal{P}$, such that the total complexity of $\mathcal{Q}$ is $O(k^2/\varepsilon^{3/2})$ and $d_{\mathcal{Q}}(s, t) \leq (1 + \varepsilon/2) \pi_{\mathcal{P}}(s, t) + \varepsilon d/8$.*

Computing ε-short paths. Let $d^* = d_{\mathcal{P}}(s, t)$. We start by computing a $2k$-factor approximation $\widetilde{d}$ to d^* in $O(n)$ time using the algorithm of Hershberger and Suri [13]. We have $d^* \leq \widetilde{d} \leq 2kd^*$. Set $m = \log(4k)$, and let $d_0 = \widetilde{d}/2k$ and $d_i = 2^i \cdot d_0$, for $i = 1, \cdots, m$. Note that $d_0 \leq d^*$ and $d_m \geq 2d^*$.

For each $i = 1, \cdots, m$, we run the algorithm of Lemma 4 with $d = d_i$ and error parameter $\varepsilon/3$ to compute a set $\mathcal{Q}_i$. We then apply Clarkson's algorithm on $\mathcal{Q}_i$ and compute an $(\varepsilon/3)$-short path π_i between s and t in $\mathcal{F}(\mathcal{Q}_i)$.[1] If $|\pi_i| \leq d_i$, then $\pi_i \subseteq C_{4d_i}$ and therefore $\pi_i \subseteq \mathcal{F}(\mathcal{P})$ (recall that while constructing $\mathcal{Q}_i$ we clipped the set $\mathcal{P}$ with the cube C_{4d_i}). This implies $d_i \geq |\pi_i| \geq d^*$, or in other words, for all $d_i < d^*$, $|\pi_i| > d_i$. On the other hand, for all $d_i \geq 2d^*$, by Lemma 4, the above procedure returns a path π_i of length at most $(1 + \varepsilon/3)((1 + \varepsilon/6)d^* + \varepsilon d_i/24 \leq d_i$. It follows that there is an index i such that, for all $j \geq i$, $|\pi_i| \leq d_i$, and for all $j < i$, $|\pi_i| > d_i$, which can be computed by a binary search on $d_1, \cdots, d_m$. For this index i, we have $d^* \leq d_i < 4d^*$ and therefore by Lemma 4,

[1] Clarkson's algorithm is divided into two steps. The first step computes an $O(1)$-short path and the second step refines this approximation to obtain an ε-short path. Since, we assume we already have a constant-factor approximation, only the refinement step is needed.

$|\pi_i| \le (1+\varepsilon/3)^2 d^* \le (1+\varepsilon)d^*$. Furthermore, $|\pi_i| \le (1+\varepsilon)d^*$ implies that $\pi_i \subseteq C_{4d_i}$ and therefore $\pi_i \subseteq \mathcal{F}(\mathcal{P})$. Hence, π_i is an ε-short path between s and t in $\mathcal{F}(\mathcal{P})$.

We spend $O(n)$ time for precomputing the Dobkin-Kirkpatrick hierarchy of each polytope in $\mathcal{P}$, and computing the value of $\widetilde{d}$. Then, in each iteration, computing $\mathcal{Q}_i$ takes $O(k^2 \log^2 k + (k^2/\varepsilon^{3/2}) \log(k/\varepsilon))$ time (Lemma 4), and running Clarkson's algorithm takes $O((k^4/\varepsilon^7) \log^2(k/\varepsilon))$ time. Since the total number of iterations is $O(\log m) = O(\log\log k)$, the total running time is $O(n+(k^4/\varepsilon^7)\log^2(k/\varepsilon)\log\log k)$. The output path has $O(k^2/\varepsilon^{3/2})$ edges, because each $\mathcal{Q}_i$ is of total complexity $O(k^2/\varepsilon^{3/2})$. Hence, we conclude the following.

Theorem 2 *Let $\mathcal{P}$ be a set of k pairwise-disjoint convex polytopes in $\mathbb{R}^3$ of total complexity n. For any two points $s, t \in \mathcal{F}(\mathcal{P})$ and a parameter $0 < \varepsilon \le 1$, an ε-short path between s and t which consists of $O(k^2/\varepsilon^{3/2})$ edges can be computed in $O(n + (k^4/\varepsilon^7)\log^2(k/\varepsilon)\log\log k)$ time.*

By combining the algorithms of Har-Peled [11] and Hershberger and Suri [13], it is possible to use the Dobkin-Kirkpatrick hierarchies to compute the value of $\widetilde{d}$ in $O(k \log n)$ time. We then obtain the following result.

Corollary 2 *Let $\mathcal{P}$ be a set of k pairwise-disjoint convex polytopes in $\mathbb{R}^3$ of total complexity n. Suppose the Dobkin-Kirkpatrick hierarchies of the polytopes in $\mathcal{P}$ are given. For any $s, t \in \mathcal{F}(\mathcal{P})$ and any $0 < \varepsilon \le 1$, an ε-short path between s and t which consists of $O(k^2/\varepsilon^{3/2})$ edges can be computed in $O(k \log n + (k^4/\varepsilon^7)\log^2(k/\varepsilon)\log\log k)$ time.*

4 Approximate Shortest-Path Queries in $\mathbb{R}^3$

In this section we study the approximate shortest-path query problem amid $\mathcal{P}$ for a fixed source in $\mathcal{F}(\mathcal{P})$ and a fixed parameter $0 < \varepsilon \le 1$. The main result is a data structure whose size and query time are independent of n. In Sections 4.1 and 4.2, we prove a few key geometric lemmas that our data structure relies on. In Section 4.3, we present the details of the data structure.

4.1 Pseudoconvex subdivisions

For a convex polytope P, an *ε-pseudoface* of P is a region $F \subseteq \partial P$ such that for any $s, t \in F$, there exist outward normals u_s and u_t of P at s and t respectively such that $\angle u_s, u_t \le \sqrt{\varepsilon}/2$. Note that an ε-pseudoface is not necessarily a connected region.

Lemma 5 *The boundary of a convex polytope P in $\mathbb{R}^3$ (resp., $\mathbb{R}^2$) can be decomposed into a collection $\mathcal{S}$ of $O(1/\varepsilon)$ (resp., $O(1/\sqrt{\varepsilon})$) ε-pseudofaces, each of which is the union of a subset of faces of P. The decomposition can be computed in $O(|P|)$ time.*

Proof: We only prove the lemma for $\mathbb{R}^3$; the case for $\mathbb{R}^2$ is simpler. Let $\mathcal{G}$ be a decomposition of $\mathbb{S}^2$ into cells of geodesic diameter at most $\sqrt{\varepsilon}/2$. It is well known that such a decomposition $\mathcal{G}$ of size $O(1/\varepsilon)$ exists, and furthermore, for any $u \in \mathbb{S}^2$, one can locate the cell $\sigma \in \mathcal{G}$ that contains u in $O(1)$ time (using floor functions). For a cell $\sigma \in \mathcal{G}$, let $S(\sigma)$ be the collection of faces of P whose outward normals fall inside σ. (To avoid ambiguity, we choose $\mathcal{G}$ such that the outward normal of each face of P is contained in the interior of a unique cell of $\mathcal{G}$.) Let $F(\sigma) = \bigcup_{f \in S(\sigma)} f$. Clearly, $F(\sigma)$ is an ε-pseudoface of P. We conclude that $\mathcal{S} = \{F(\sigma) \mid \sigma \in \mathcal{G}\}$ is the desired collection of ε-pseudofaces of P, and clearly it can be computed in $O(|P|)$ time. □

For a set $\mathcal{P}$ of convex polytopes, we define an *ε-pseudoconvex region* in $\mathcal{F}(\mathcal{P})$ as a region $\sigma \subseteq \mathcal{F}(\mathcal{P})$ such that for any $s, t \in \sigma$, $d_{\mathcal{P}}(s,t) \le (1+\varepsilon)\|st\|$. Again, an ε-pseudoconvex region is not necessarily connected. We define an *ε-pseudoconvex decomposition* Ξ of $\mathcal{F}(\mathcal{P})$ as a decomposition of $\mathcal{F}(\mathcal{P})$ into ε-pseudoconvex regions.

A *wedge* W is the intersection of two halfspaces $h_1^+ \cap h_2^+$, and the *angle* of W is $\angle u_1, u_2$, where u_1, u_2 are the outward normals of h_1^+ and h_2^+ respectively. Let W be a wedge of angle at most $\sqrt{\varepsilon}/2$. A standard calculation shows that $d_P(s,t) \le (1+\varepsilon)\|st\|$ for any $s, t \in \mathcal{F}(W)$.

Lemma 6 *Let P be a convex polytope in $\mathbb{R}^3$. An ε-pseudoconvex decomposition $\Xi(P)$ of $\mathcal{F}(P)$ of size $O(1/\varepsilon)$ can be computed in $O(|P|)$ time.*

Proof: Let Q be the vertical projection of P onto the xy-plane, and let C be the (infinite) vertical prism $Q \times \mathbb{R}$. We produce an ε-pseudoconvex decomposition of $\mathcal{F}(P)$ by first constructing an ε-pseudoconvex decomposition Ξ' of size $O(1/\varepsilon)$ within $\mathcal{F}(P) \cap C = C \setminus \operatorname{int} P$, and then constructing an ε-pseudoconvex decomposition Ξ'' of $\mathcal{F}(P) \setminus \operatorname{int} C = \mathcal{F}(C)$ of size $O(1/\sqrt{\varepsilon})$.

The region $C \setminus P$ consists of two components: the upper component C^+ extending infinitely in $(+z)$-direction, and the lower component C^- extending infinitely in $(-z)$-direction. For a face f of P in C^+, let σ_f be the vertical prism with base f and extending infinitly in $(+z)$-direction. For an ε-pseudoface $F \in \mathcal{S}$, where $\mathcal{S}$ is defined in Lemma 5, let $\sigma_F = \bigcup_{f \subseteq C^+ \cap F} \sigma_f$. We claim that σ_F is an ε-pseudoconvex region. To see this, for any $s, t \in \sigma_F$, let s', t' be their projections onto P in $(-z)$-direction. Since s', t' lie in the same ε-pseudoface F, there are supporting planes $h_{s'}$ and $h_{t'}$ of P at s' and t' with outer normals $u_{s'}$ and $u_{t'}$, respectively, such that $\angle u_{s'}, u_{t'} \le \sqrt{\varepsilon}/2$. Let W be the wedge $h_{s'}^+ \cap h_{t'}^+$, where $h_{s'}^+$ (resp., $h_{t'}^+$) denotes the halfspaced bounded by $h_{s'}$ (resp., $h_{t'}$) and containing P. Observe that $P \subseteq W$ and $s, t \in \mathcal{F}(W)$. Since the angle of W is at most $\sqrt{\varepsilon}/2$, we then have $d_P(s,t) \le d_W(s,t) \le (1+\varepsilon)\|st\|$ as desired. It follows that $\{\sigma_F \mid F \in \mathcal{S}\}$ is an ε-pseudoconvex decomposition of C^+. Symmetrically, we can construct an ε-pseudoconvex decomposition of C^-. They together form an ε-pseudoconvex decomposition Ξ' of $C \setminus \operatorname{int} P$ of size $O(1/\varepsilon)$.

In the xy-plane, using a similar method, we can construct an ε-pseudoconvex decomposition $\Xi(Q)$ of $\mathcal{F}(Q)$ of size $O(1/\sqrt{\varepsilon})$; in particular, for any $\sigma \in \Xi(Q)$ and any $s, t \in \sigma$, there exists a wedge $W_{s,t}$ of angle at most $\sqrt{\varepsilon}/2$ in the xy-plane such that $Q \subseteq W_{s,t}$ and $s, t \in \mathcal{F}(W_{st})$. For any $s, t \in \sigma \times \mathbb{R}$ with $\sigma \in \Xi(Q)$, let s', t' be the projection of s, t onto σ. Then $W = W_{s't'} \times \mathbb{R}$ is a wedge of angle at most $\sqrt{\varepsilon}/2$ such that $P \subseteq W$ and $s, t \in \mathcal{F}(W)$, implying $d_P(s,t) \le\le d_W(s,t) \le (1+\varepsilon)\|st\|$. As such, $\sigma \times \mathbb{R}$ is an ε-pseudoconvex region, and the set $\Xi'' = \{\sigma \times \mathbb{R} \mid \sigma \in \Xi(Q)\}$ is an ε-pseudoconvex decomposition of $\mathcal{F}(C)$. We conclude that $\Xi(P) = \Xi' \cup \Xi''$ is an ε-pseudoconvex decomposition of $\mathcal{F}(P)$ of size $O(1/\varepsilon)$. □

Using Lemma 6, we construct an ε-pseudoconvex decomposition of $\mathcal{F}(\mathcal{P})$ as follows. For $i = 1, \cdots, k$, set $P_i' = \bigcap_{j \ne i} h_{i,j}^+$, where $h_{i,j}$ is a plane separating P_i and P_j and $h_{i,j}^+$ is the halfspace bounded by $h_{i,j}$ and containing P_i. Clearly, P_i' is a convex polytope of complexity $O(k)$ with $P_i \subseteq P_i'$, and $\mathcal{P}' = \{P_1', P_2', \ldots, P_k'\}$ is a set of pairwise-disjoint convex polytopes. We decompose $\mathcal{F}(\mathcal{P}')$ into a set Ξ_0 of $O(k^3 \log k)$ tetrahedra as described in [1]; each tetrahedron is clearly an ε-pseudoconvex region. Next, for each polytope P_i, we obtain an ε-pseudoconvex decomposition $\Xi(P_i)$ of $\mathcal{F}(P_i)$ from Lemma 6, and clip each region $\sigma \in \Xi(P_i)$ with P_i'. Let $\Xi_i = \{\sigma \cap P_i' \mid \sigma \in \Xi(P_i)\}$ denote the resulting decomposition of $P_i' \setminus \operatorname{int} P_i$. (We remark that, for our purpose, there is no need to represent each cell $\sigma \cap P_i'$ explicitly.) Each region $\sigma \cap P_i' \in \Xi_i$ is an ε-pseudoconvex region. This is because, for any pair of points $s, t \in \sigma \cap P_i'$, $\pi_{\mathcal{P}}(s,t) \subseteq P_i' \setminus \operatorname{int} P_i$, implying $d_{\mathcal{P}}(s,t) = d_P(s,t) \le (1+\varepsilon)\|st\|$. Setting $\Xi(\mathcal{P}) = \Xi_0 \cup \Xi_1 \cup \cdots \cup \Xi_k$, we obtain the following.

Lemma 7 *$\Xi(\mathcal{P})$ is an ε-pseudoconvex decomposition of $\mathcal{F}(\mathcal{P})$ of size $O(k^3 \log k + k/\varepsilon)$.*

4.2 Critical distance values

Let s be a fixed source in $\mathcal{F}(\mathcal{P})$. For a region $U \subseteq \mathcal{F}(\mathcal{P})$, we call a set of distance values $d_1 < \cdots < d_m$ *critical* if for any $t \in U$, one of the following holds:

(i) $d_{\mathcal{P}}(s,t) \le (1+\varepsilon)\|st\|$; or
(ii) there exists an index i such that $d_i \le d_{\mathcal{P}}(s,t) \le d_{i+1} \le 2d_i$.

Intuitively, the critical distance values of U are what we need to focus on when answering approximate shortest-path queries for points in U; for other cases the Euclidean distances would already be a good approximation. We next describe an algorithm to compute a set $\Sigma(P_i)$ of $O((1/\varepsilon)\log(1/\varepsilon))$ critical distance values for the region $P_i' \setminus \operatorname{int} P_i$, for each $i = 1, \cdots, k$.

Let Ξ_i be an $(\varepsilon/4)$-pseudoconvex decomposition of $P_i' \setminus \operatorname{int} P_i$ of size $O(1/\varepsilon)$. We will compute a set Σ_σ of critical distance values for for each $(\varepsilon/4)$-pseudoconvex region $\sigma \in \Xi_i$ and then set $\Sigma(P_i) = \bigcup_{\sigma\in\Xi_i}\Sigma_\sigma$. The set Σ_σ is computed as follows. We first find the Euclidean nearest neighbor v_e of s in σ (i.e., $v_e = \arg\min_{p\in\Xi_i}\|sp\|$) using a method to be explained shortly. Using Corollary 2, we compute a value $\tilde{r}$ such that $d_{\mathcal{P}}(s,v_e) \le \tilde{r} \le 2d_{\mathcal{P}}(s,v_e)$. We then set

$$\Sigma_\sigma = \left\{\tilde{r}/8, 2\tilde{r}/8, 2^2\tilde{r}/8, \ldots, 2^{m+3}\tilde{r}/8\right\},$$

where $m = \lceil \log_2(4+4/\varepsilon)\rceil$.

To compute the nearest neighbor v_e of s in σ, recall that $\sigma = \sigma' \cap P_i'$ for some $\sigma' \in \Xi(P_i)$, where $\Xi(P_i)$ is an $(\varepsilon/4)$-pseudoconvex decomposition of $\mathcal{F}(P_i)$ from Lemma 6. Using the Dobkin-Kirkpatrick hierarchy of P_i', one can compute the Euclidean nearest neighbor of s in $\sigma' \cap P_i'$ (i.e., v_e) in $O(|\sigma'|\log|P_i'|) = O(|\sigma'|\log k)$ time, where $|\sigma'|$ denotes the complexity of the cell σ'.

Once v_e is identified, the value $\tilde{r}$ and thus Σ_σ can be computed in $O(k\log n + k^4\log^2 k\log\log k + |\sigma'|\log k)$ time using Corollary 2 (after linear-time preprocessing). Since $\sum_{\sigma'\in\Xi_i}|\sigma'| = O(|P_i|)$ and $|\Xi_i| = O(1/\varepsilon)$, $\Sigma(P_i)$ can then be computed in $O((k/\varepsilon)\log n + (k^4/\varepsilon)\log^2 k\log\log k + |P_i|\log k)$ time.

It remains to prove that Σ_σ is indeed a set of critical distance values for the region σ. Let $v_g \in \sigma$ be the geodesic nearest neighbor of s in σ (i.e., $v_g = \arg\min_{p\in\sigma} d_{\mathcal{P}}(s,p)$), and let $r = d_{\mathcal{P}}(s,v_g)$. Since σ is an $(\varepsilon/4)$-pseudoconvex region and $v_e, v_g \in \sigma$, we have

$$d_{\mathcal{P}}(v_e,v_g) \le (1+\varepsilon/4)\|v_e v_g\| \le (1+\varepsilon/4)(\|sv_e\| + \|sv_g\|) \le 2(1+\varepsilon/4)d_{\mathcal{P}}(s,v_g).$$

It follow that $d_{\mathcal{P}}(s,v_e) \le d_{\mathcal{P}}(s,v_g) + d_{\mathcal{P}}(v_g,v_e) \le 4d_{\mathcal{P}}(s,v_g)$, implying $r \le \tilde{r} \le 8r$.

Next, we partition the region σ into two subsets: $\sigma_1 = \{t \in \sigma \mid \|v_g t\| \ge r(1+4/\varepsilon)\}$ and $\sigma_2 = \{t \in \sigma \mid \|v_g t\| \le r(1+4/\varepsilon)\}$. For any point $t \in \sigma_1$, we have $r \le \|v_g t\|/(1+1/\varepsilon)$. Hence,

$$\|st\| \ge \|v_g t\| - \|sv_g\| \ge \|v_g t\| - r \ge \|v_g t\|(1 - 1/(1+4/\varepsilon)) = \|v_g t\|/(1+\varepsilon/4).$$

Furthermore,

$$d_{\mathcal{P}}(s,t) \le d_{\mathcal{P}}(s,v_g) + d_{\mathcal{P}}(v_g,t) \le r + (1+\varepsilon/4)\|v_g t\| \le (1+3\varepsilon/4+\varepsilon^2/16)\|v_g t\|/(1+\varepsilon/4).$$

Therefore, $d_{\mathcal{P}}(s,t) \le (1+3\varepsilon/4+\varepsilon^2/16)\|st\| \le (1+\varepsilon)\|st\|$.

On the other hand, for any point $t \in \sigma_2$,

$$r \le d_{\mathcal{P}}(s,t) \le r + (1+\varepsilon/4)\|v_g t\| \le (1+(1+\varepsilon/4)(1+4/\varepsilon))r \le (4+4/\varepsilon)r.$$

Together with $r \le \tilde{r} \le 8r$, we obtain $\tilde{r}/8 \le d_{\mathcal{P}}(s,t) \le (4+4/\varepsilon)\tilde{r}$. Therefore, there exists an index $0 \le i < m+3$ such that $2^i\tilde{r}/8 \le d_{\mathcal{P}}(s,t) \le 2^{i+1}\tilde{r}/8$, as desired.

Lemma 8 *A set $\Sigma(P_i)$ of $O((1/\varepsilon)\log(1/\varepsilon))$ critical distance values for the region $P_i' \setminus \operatorname{int} P_i$ can be computed in $O((k/\varepsilon)\log n + (k^4/\varepsilon)\log^2 k\log\log k + |P_i|\log k)$ time.*

4.3 Data structures

We are now ready to describe the data structure for approximate shortest-path queries with respect to a fixed source $s \in \mathcal{F}(\mathcal{P})$. We first present a simpler one that only reports a distance value, and then describe the necessary changes to it so that an ε-short path can also be reported.

The structure. Recall from Sections 4.1 and 4.2 that $\mathcal{P}'$ is a set of k pairwise-disjoint convex polytopes, each of complexity $O(k)$, such that for each $P \in \mathcal{P}$ there exists $P' \in \mathcal{P}'$ with $P \subseteq P'$, Ξ_0 is the decomposition of $\mathcal{F}(\mathcal{P}')$ into $O(k^3 \log k)$ tetrahedra, and $\Sigma(P)$ is a set of $m = O((1/\varepsilon)\log(1/\varepsilon))$ critical distance values for the region $P' \setminus \operatorname{int} P$.

For each tetrahedron $\Delta \in \Xi_0$, we construct a data structure $\mathbb{D}(\Delta)$ of Har-Peled [12] of size $O(1/\varepsilon^5)$ in $O(1/\varepsilon^5)$ time so that for any point $t \in \Delta$, an ε-short distance between s and t amid $\mathcal{P}$ can be reported in $O(\log(1/\varepsilon))$ time. The data structure of [12] in a tetrahedron Δ is constructed by sprinkling a set of $O((1/\varepsilon^2)\log(1/\varepsilon))$ weighted points in Δ and then computing a weighted Voronoi diagram of this point set togother with a point-location structure on top of it; the weight w_p of each weighted point p is an ε-short distance between s and p amid $\mathcal{P}$. To answer a query for a query point $t \in \Delta$, one computes the weighted nearest neighbor q of t among the weighted points and returns the distance value $w_q + \|qt\|$.

For each $P \in \mathcal{P}$ and each $d \in \Sigma(P)$, we construct a data structure $\mathbb{D}(P, d)$ as follows. Set $r = \varepsilon^{3/2} d/c$ for a sufficiently large constant $c > 0$. We compute an *inner r-approximation* I of $P \cap C_{4d}$ so that $I \subseteq P \cap C_{4d} \subseteq I_r$ and $|I| = O(1/\varepsilon^{3/2})$. We then decompose the region $(P' \cap C_{4d}) \setminus \operatorname{int} I$ into $O(|P' \cap C_{4d}| + |I|) = O(k + 1/\varepsilon^{3/2})$ tetrahedra using an algorithm in [6], and process them into a point-location structure of size $O((k + 1/\varepsilon^{3/2})\log^2(k/\varepsilon))$ with query time $O(\log^2(k/\varepsilon))$ [19]. For each tetrahedron Δ, we again construct a data structure of Har-Peled [12] so that for any point $t \in \Delta$, an ε-short distance between s and t amid $(\mathcal{P} \setminus \{P\}) \cup \{I\}$ can be reported in $O(\log(1/\varepsilon))$ time (in this data structure, the weight w_p of each sprinkled point $p \in \Delta$ is an ε-short distance between s and p amid $(\mathcal{P} \setminus \{P\}) \cup \{I\}$). In summary, the data structure $\mathbb{D}(P, d)$ can be used to report, for any $t \in (P' \cap C_{4d}) \setminus \operatorname{int} I$, an ε-short distance between s and t amid $(\mathcal{P} \setminus \{P\}) \cup \{I\}$ in $O(\log^2(k/\varepsilon))$ time.

Finally, we preprocess Ξ_0 and $\mathcal{P}'$ into a point-location data structure of size $O(k^3 \log^3 k)$ so that given a point $t \in \mathcal{F}(\mathcal{P})$, one of $\Delta \in \Xi_0$ or $P' \in \mathcal{P}'$ that contains t can be located in $O(\log^2 k)$ time [19].

Using Lemma 8, we can compute the critical distance values in $\Sigma(P_1) \cup \cdots \cup \Sigma(P_k)$ in $O(n \log k + (k^5/\varepsilon)\log^2 k \log\log k)$ time. The weight of each sprinkled point in Har-Peled's data structure can be computed in $O(k \log n + (k^4/\varepsilon^7)\log^2(k/\varepsilon)\log\log k)$ time using Corollary 2 (after $O(n)$-time preprocessing). Therefore, the total time for constructing the entire data structure (which is dominated by the time for computing all the weights plus the time for computing the critical distance values) is

$$O\left(n \log k + \left(k^2 m + km/\varepsilon^{3/2} + k^3 \log k\right)\left((1/\varepsilon^2)\log(1/\varepsilon)\right)\left(k \log n + (k^4/\varepsilon^7)\log^2(k/\varepsilon)\log\log k\right)\right).$$

The size of the entire data structure is

$$O\left((k + 1/\varepsilon^{3/2})(\log^2 k + 1/\varepsilon^5) + k^3 \log k/\varepsilon^5 + k^3 \log^3 k\right).$$

Query algorithm. To answer an approximate shortest-path query for a query point $t \in \mathcal{F}(\mathcal{P})$, we proceed as follows. If t is contained in some tetrahedron $\Delta \in \Xi_0$, then we simply use $\mathbb{D}(\Delta)$ to report an ε-short distance between s and t and are done. Otherwise, t is contained in a polytope $P' \in \mathcal{P}'$ for some $P \in \mathcal{P}$. Let $d_1 < \cdots < d_m$ be the set of values in $\Sigma(P)$, where $m = O((1/\varepsilon)\log(1/\varepsilon))$. For each d_j, let I_j be the aforementioned inner r_j-approximation of $P \cap C_{4d_j}$ with $r_j = \varepsilon^{3/2} d_j/c$, and let $\widetilde{d}_j$ be the distance value returned by querying t against the data structure $\mathbb{D}(P, d_j)$. We find an index

i in $\{j \mid d_j \geq \|st\|\}$ such that $\widetilde{d}_i \leq (1+\varepsilon)d_i$ and $\widetilde{d}_{i-1} > (1+\varepsilon)d_{i-1}$; this index can be computed by a binary search. We then report $\max\left\{(1+\varepsilon)\widetilde{d}_i, (1+\varepsilon)\|st\|\right\}$.

Lemma 9 $d_{\mathcal{P}}(s,t) \leq \max\left\{(1+\varepsilon)\widetilde{d}_i, (1+\varepsilon)\|st\|\right\} \leq (1+3\varepsilon)d_{\mathcal{P}}(s,t)$.

Proof: Set $d^* = d_{\mathcal{P}}(s,t)$. First note that for any j, we have

$$\widetilde{d}_j \leq (1+\varepsilon)d_{(\mathcal{P}\setminus\{P\})\cup\{I_j\}}(s,t) \leq (1+\varepsilon)d^*. \tag{3}$$

Furthermore, for any d_j such that $d_j \geq \|st\|$, using the same method to prove (1) of Section 3, it can be shown that

$$d^* \leq (1+\varepsilon/2)d_{(\mathcal{P}\setminus\{P\})\cup\{I_j\}}(s,t) + \varepsilon d_j/8. \tag{4}$$

Next we distinguish two cases (recall that $\Sigma(P)$ is a set of critical distance values for the region $(\mathcal{P} \setminus \{P\}) \cup \{I\}$):

Case 1. $d^* \leq (1+\varepsilon)\|st\|$. Combined with (3), clearly $\max\left\{(1+\varepsilon)\widetilde{d}_i, (1+\varepsilon)\|st\|\right\}$ is a (3ε)-short distance between s and t in $\mathcal{F}(\mathcal{P})$.

Case 2. There exists an index j such that $d_j \leq d^* \leq d_{j+1} \leq 2d_j$. By (3), we know that $d_{i-1} \leq d^*$. Since $d_{i-1} \leq d^*$ and $d_j \leq d^*$, we have $d_i \leq d_{j+1} \leq 2d^*$. Therefore by (4),

$$\widetilde{d}_i \geq d_{(\mathcal{P}\setminus\{P\})\cup\{I_i\}}(s,t) \geq (d^* - \varepsilon d_i/8)/(1+\varepsilon/2) \geq (1-\varepsilon/4)d^*/(1+\varepsilon/2).$$

As such, $(1+\varepsilon)\widetilde{d}_i \geq d^*$. Together with (3), we conclude that $(1+\varepsilon)\widetilde{d}_i$, and subsequently $\max\left\{(1+\varepsilon)\widetilde{d}_i, (1+\varepsilon)\|st\|\right\}$, is a (3ε)-short distance between s and t in $\mathcal{F}(\mathcal{P})$.

The lemma then follows. □

After rescaling ε, we obtain the following theorem.

Theorem 3 *Let $\mathcal{P}$ be a set of k convex polytopes of total complexity n in $\mathbb{R}^3$, and let s be a fixed source in $\mathcal{F}(\mathcal{P})$. For any fixed parameter $0 < \varepsilon \leq 1$, a data structure of size $O(k^3 \operatorname{poly}(\log k, 1/\varepsilon))$ can be constructed in $O(n \log k + k^7 \operatorname{poly}(\log k, 1/\varepsilon))$ time such that for any query point $t \in \mathcal{F}(\mathcal{P})$, an ε-short distance between s and t can be reported in $O(\log^2(k/\varepsilon)\log(1/\varepsilon))$ time.*

Remark. By a combined use of inner and outer approximations of polytopes, we are able to modify the above data structure so that it also reports an ε-short path between s and t in $O(k^2/\varepsilon^{3/2})$ time. The details are presented in the appendix.

Theorem 4 *Let $\mathcal{P}$ be a set of k convex polytopes of total complexity n in $\mathbb{R}^3$, and let s be a fixed source in $\mathcal{F}(\mathcal{P})$. For any fixed parameter $0 < \varepsilon \leq 1$, a data structure of size $O(k^5 \operatorname{poly}(\log k, 1/\varepsilon))$ can be constructed in $O(n \log k + k^7 \operatorname{poly}(\log k, 1/\varepsilon))$ time such that for any query point $t \in \mathcal{F}(\mathcal{P})$, an ε-short distance between s and t can be reported in $O(\log^2(k/\varepsilon)\log(1/\varepsilon))$ time. An ε-short path of complexity $O(k^2/\varepsilon^{3/2})$ between s and t can be reported in an additional $O(k^2/\varepsilon^{3/2})$ time.*

References

[1] P. K. Agarwal, B. Aronov, and S. Suri. Stabbing triangulations by lines in 3d. In *Proc. 11th Annu. Sympos. Comput. Geom.*, pages 267–276, 1995.

[2] P. K. Agarwal, S. Har-Peled, and M. Karia. Computing approximate shortest paths on convex polytopes. In *Symposium on Computational Geometry*, pages 270–279, 2000.

[3] P. K. Agarwal, S. Har-peled, M. Sharir, and K. R. Varadarajan. Approximating shortest paths on a convex polytope in three dimensions. *J. ACM*, 44:567–584, 1997.

[4] T. Asano, D. G. Kirkpatrick, and C. Yap. Pseudo approximation algorithms with applications to optimal motion planning. *Discrete Comput. Geom.*, 31:139–171, 2004.

[5] J. Canny and J. Reif. New lower bound techniques for robot motion planning problems. In *Proc. 28th IEEE Sympos. Foundat. Comput. Sci.*, pages 49–60, 1987.

[6] B. Chazelle and N. Shouraboura. Bounds on the size of tetrahedralizations. *Discrete Comput. Geom.*, 14:429–444, 1995.

[7] J. Chen and Y. Han. Shortest paths on a polyhedron, part i: Computing shortest paths. *Internat. J. Comput. Geom. Appl.*, 6:127–144, 1996.

[8] J. Choi, J. Sellen, and C.-K. Yap. Approximate Euclidean shortest paths in 3-space. *Internat. J. Comput. Geom. Appl.*, 7:271–295, 1997.

[9] K. L. Clarkson. Approximation algorithms for shortest path motion planning. In *Proc. 19th Annu. ACM Sympos. Theory Comput.*, pages 56–65, 1987.

[10] K. Dudley. Metric entropy of some classes of sets with differentiable boundaries. *J. Approx. Theory*, 10:227–236, 1974.

[11] S. Har-Peled. Approximate shortest-path and geodesic diameter on convex polytopes in three dimensions. *Discrete Comput. Geom.*, 21:216–231, 1999.

[12] S. Har-Peled. Constructing approximate shortest path maps in three dimensions. *SIAM J. Comput.*, 28:1182–1197, 1999.

[13] J. Hershberger and S. Suri. Practical methods for approximating shortest path on a convex polytope in $\mathbb{R}^3$. *Comput. Geom. Theory Appl.*, 10:31–46, 1998.

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

[15] J. Mitchell. Shortest paths and networks. In Jacob E. Goodman and Joseph O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 755–778. CRC Press, Inc., Boca Raton, FL, USA, 1997.

[16] J.S.B. Mitchell, D. Mount, and C. Papadimitriou. The discrete geodesic problem. *SIAM J. Comput.*, 16:647–668, 1987.

[17] C. Papadimitriou. An algorithm for shortest path motion planning in three dimension. *Inform. Process. Lett.*, 20:259–268, 1985.

[18] A. Pogorelov. *Extrinsic Geometry of Convex Surfaces*, volume 35 of *Transactions of Mathematical Monographs.* American Mathematical Society, Providence, RI, 1973.

[19] F. Preparata and R. Tamassia. Efficient point location in a convex spatial cell-complex. *SIAM J. Comput.*, 21:267–280, 1992.

[20] J. Reif and J. Storer. A single-exponential upper bound for finding shortest paths in three dimensions. *J. ACM*, 41:1013–1019, 1994.

[21] Y. Schreiber and M. Sharir. An optimal algorithm for shortest paths on a convex polytope in three dimensions. *Discrete Comput. Geom.*, 39:500–579, 2008.

[22] M. Sharir. On shortest paths amidst convex polyhedra. *SIAM J. Comput.*, 16:561–572, 1987.

[23] M. Sharir and A. Schorr. On shortest paths in polyhedral spaces. *SIAM J. Comput.*, 15:193–215, 1986.

Proof of Lemma 3. Let p' and q' be the closest points of p and q on ∂P respectively. Let $h_{p'}$ and $h_{q'}$ be the planes passing through p' and q' with normals in the direction $p'p$ and $q'q$ respectively. It is easy to see that $h_{p'}$ and $h_{q'}$ are supporting planes of P. Let p'' (resp., q'') be the intersection of ∂P_r with the ray emanating from p' (resp., q') in direction $p'p$ (resp. $q'q$). Observe that $\|p'p''\| = \|q'q''\| = r$. Furthermore, since $P \subseteq P' \subseteq P_r$, the segment $\overline{p'p''}$ contains p and the segment $\overline{q'q''}$ contains q.

It has been shown in [3] that

$$d_{P'}(p'', q'') \leq (1 + \varepsilon/2) d_P(p', q') + 2\pi r + 2r + 100r/\sqrt{\varepsilon}.$$

Furthermore, observe that $d_{P'}(p, q) \leq \|pp''\| + d_{P'}(p'', q'') + \|q''q\|$ and $d_P(p', q') \leq \|p'p\| + d_P(p, q) + \|qq'\|$. Putting these inequalities together, we obtain (2) as claimed.

Reporting an ε-short path. We now modify the data structure of Section 4.3 slightly so that it is also able to report an ε-short path between s and the query point t. For each $P \in \mathcal{P}$ and each $d \in \Sigma(P)$, we redefine $\mathbb{D}(P, d)$ as follows. Set $r = \varepsilon^{3/2} d/c$ for a sufficiently large constant $c > 0$. We compute an *outer r-approximation* O of $P \cap C_{4d}$ so that $P \cap C_{4d} \subseteq O \subseteq (P \cap C_{4d})_r$ and $|O| = O(1/\varepsilon^{3/2})$. We then decompose the region $(P' \cap C_{4d}) \setminus \operatorname{int} O$ into $O(k + 1/\varepsilon^{3/2})$ tetrahedra and process them into a point-location structure. For each tetrahedron Δ, we construct a data structure of [12]. We also compute an inner r-approximation I of $P \cap C_{4d}$ as before and precompute the Dobkin-Kirkpatrick hierarchies of I and O.

In addition, recall that the data structure of Har-Peled [12] in each tetrahedron is a weighted Voronoi diagram of a set of $O((1/\varepsilon^2)\log(1/\varepsilon))$ weighted points. We store an ε-short path of complexity $O(k^2/\varepsilon^{3/2})$ from s to each of these points (see Corollary 2). The rest of the data structure remains the same. The size of the data structure becomes $O(k^5 \operatorname{poly}(\log k, 1/\varepsilon))$, while the construction time remains the same.

To answer a query, we proceed almost the same as before. The only difference occurs when the query point t lies inside $O \setminus \operatorname{int} I$, where O and I is the outer and inner approximations of $P \cap C_{4d}$ for some $P \in \mathcal{P}$ and $d \in \Sigma(P)$. In this case, we first compute the Euclidean nearest neighbor t' of t on I using the Dobkin-Kirkpatrick hierarchy of I, and then compute the project t'' of t onto ∂O along direction $t't$ using the Dobkin-Kirkpatrick hierarchy of O. Since $t'' \in (P \cap C_{4d}) \setminus \operatorname{int} O$, we can then query $\mathbb{D}(P, d)$ with t''. Recall that the query process identifies the weighted nearest neighbor q of t'' among the weighted points belonging to the tetrahedron containing t''. Let π be the precomputed path from s to q. We then return the concatenation of π, qt'', and $t''t$ as the path from s to t, and return $|\pi| + \|qt''\| + \|t''t\|$ as the distance value.

The correctness of the above procedure follows from the same arguments for Theorem 3 and the observation that tt'' is a line segment lying completely inside $\mathcal{F}(\mathcal{P})$ and $\|tt''\| \leq 2r = 2\varepsilon^{3/2} d/c$. Theorem 4 then follows.