Approximation Algorithms for Shortest Path Motion Planning

extended abstract

Kenneth L. Clarkson
AT&T Bell Laboratories
Murray Hill, New Jersey 07974

Abstract

This paper gives approximation algorithms for solving the following motion planning problem: Given a set of polyhedral obstacles and points s and t, find a shortest path from s to t that avoids the obstacles. The paths found by the algorithms are piecewise linear, and the length of a path is the sum of the lengths of the line segments making up the path. Approximation algorithms will be given for versions of this problem in the plane and in three-dimensional space. The algorithms return an ϵ-short path, that is, a path with length within $(1+\epsilon)$ of shortest. Let n be the total number of faces of the polyhedral obstacles, and ϵ a given value satisfying $0 < \epsilon \leq \pi$. The algorithm for the planar case requires $O(n \log n)/\epsilon$ time to build a data structure of size $O(n/\epsilon)$. Given points s and t, an ϵ-short path from s to t can be found with the use of the data structure in time $O(n/\epsilon + n \log n)$. The data structure is associated with a new variety of Voronoi diagram. Given obstacles $S \subset E^3$ and points $s, t \in E^3$, an ϵ-short path between s and t can be found in

$$O(n^2 \lambda(n) \log(n/\epsilon)/\epsilon^4 + n^2 \log n\rho \log(n \log \rho))$$

time, where ρ is the ratio of the length of the longest obstacle edge to the distance between s and t. The function $\lambda(n) = \alpha(n)^{O(\alpha(n)^{O(1)})}$, where the $\alpha(n)$ is a form of inverse of Ackermann's function. For $\log(1/\epsilon)$ and $\log \rho$ that are $O(\log n)$, this bound is $O(n^2 (\log^2 n) \lambda(n)/\epsilon^4)$.

1 Introduction

1.1 Results and related work

Motion planning is the problem of determining a path by which an object can be moved from place to place while avoiding obstacles. A survey of the substantial literature on various cases of this problem can be found in [Yap85]. This paper describes algorithms for the problem of moving in a short path from one point to another in a way that avoids given polyhedral obstacles.

In the case where the obstacles, points, and paths are confined to the plane, previous work has generally focused on finding exact algorithms: those finding a shortest path between two points. This work has shown that $O(n^2 \log n)$ time suffices for this problem[SS84], and with sophisticated improvements, an $O(n^2)$ bound on time and space can be attained [AAG$^+$85]. Recently, Chew has shown [Che86] that $O(n^2)$ time and $O(n)$ space are enough to find a $(\sqrt{10}-1)$-short path. In this paper, an algorithm is given that requires $O(n \log n/\epsilon)$ time to build a data structure of size $O(n/\epsilon)$, so that given two points, in $O(n/\epsilon)$ time an ϵ-short path between the two points can be found. Indeed, with $O(n^2+n \log n/\epsilon)$ time and space, a data structure can be built so that an ϵ-short path can be found in $O((\log n)/\epsilon)$ time.

The general three-dimensional version of this problem was posed in [SS84]. All known exact algorithms for this version require exponential time, although some polynomial-time algorithms are known for special cases [Mou84, SB86, RS85]. As for approximation algorithms, Papadimitriou [Pap85] has shown that $O(n^3(L+\log(n/\epsilon))^2/\epsilon)$ time suffices to find an ϵ-short path between two points. Here L is a bound on the number of bits in an integer specifying the coordinates of a point in an instance. (In the worst case, $\log \rho = \Theta(L)$, but in this paper $\log \rho$ is used, as in many cases, L may be large but $\log \rho$ small.) The algorithm given in this paper is faster than that of [Pap85] when $n\epsilon^3$ is large. Also, the number of arithmetic operations that are done depends much less on $\log \rho = \Theta(L)$. Another contrasting feature is that the new algorithm manipulates only "flat" sets defined by sets of linear inequalities, where the algorithm of [Pap85] manipulates regions bounded by hyperbolas.

1.2 Outline of the paper

This section continues with an overview and preliminary definitions. In §2, the subgraph $\mathcal{V}_\epsilon$ is formally defined, and its properties related to ϵ-short paths in the plane are proven. In §3, an algorithm is given for constructing $\mathcal{V}_\epsilon$ in the planar case. In §4, the three-dimensional version of $\mathcal{V}_\epsilon$ is introduced, and in §5, an algorithm is given for computing ϵ-short paths in E^3. In §6, generalization of the results to arbitrary l_p norms is considered.

1.3 The general idea

In this section, an overview will be given of the ideas used in this paper, with intuitive justifications for the main construction.

Let S be a set of polygonal obstacles, so that we are interested in paths in the plane that do not intersect the interior of S. In previous work, it has been shown that a shortest such path from a point s to a point t has endpoints are that either s or t, or are in the set $\operatorname{vert} S$ of vertices of S [SS84]. To determine a shortest path exactly, it is useful to construct the *visibility graph*: the undirected graph $\mathcal{V}$ whose vertex set is $\operatorname{vert} S$, and with an edge $x \overset{\mathcal{V}}{-\!\!-} y$ between vertices x and y if and only if x and y are visible to each other, that is, the line segment $\overline{xy}$ does not intersect the interior of S. The number of edges in $\mathcal{V}$ is $\Theta(n^2)$, where $n = |\operatorname{vert} S|$. Suppose $\mathcal{V}$ is augmented by adding s and t as vertices, and also adding corresponding edges. Also, let every edge of $\mathcal{V}$ be weighted by the distance between its endpoints. Then an algorithm for the single-source path problem on graphs can be applied, and the shortest path tree from s may be found in $O(n^2)$ time. This tree has the property that the path in the tree from s to each vertex is a shortest path. Given the visibility graph $\mathcal{V}$, a shortest path from s to t may therefore be found in $\Theta(n^2)$ time.

In order to obtain a faster algorithm for short path motion planning, one useful step is to eliminate unnecessary edges from $\mathcal{V}$, and find a subgraph $\mathcal{V}_\epsilon$ of $\mathcal{V}$ such that a shortest path in $\mathcal{V}_\epsilon$ from one vertex to another is a short path in $\mathcal{V}$. For this approach to be useful, $\mathcal{V}_\epsilon$ should have few edges.

Chew has applied this approach to motion planning [Che86], and it has been previously been applied to the geometric minimum spanning tree problem [Yao82].

In the geometric minimum spanning tree problem, a set S of n points is given in the d-dimensional Euclidean space E^d, and a minimum spanning tree (MST) is desired for the weighted undirected graph G defined for those points: the vertices of the graph are the points, and for each pair of points, there is an edge in G, weighted by the distance between the points. Here again, application of algorithms for general weighted graphs gives a $\Theta(n^2)$ algorithm for this problem. It was shown by A. Yao that to obtain an MST of this graph G, it is enough to find a minimum spanning tree of a certain subgraph $\check{G}$, obtained as follows: for every point $a \in S$, consider the 12 cones around a, as shown in Figure 1. For each such cone C, include in $\check{G}$ the edge from a to the closest point in $S \cap C$. The resulting subgraph has at most $12n$ edges, and contains a minimum spanning tree of G. Therefore, if $\check{G}$ can be obtained quickly, an MST can be found in $o(n^2)$ time.

Another fact holds true about this subgraph $\check{G}$: for any $a, b \in S$, the total length of a path $a \overset{\check{G}}{-\!\!-\!\!-} b$ from a to b is at most a factor of $1 + \sqrt{3}$ larger than the distance between a and b. That is, a shortest path in $\check{G}$ is an ϵ-short path in G.

Intuitively, this fact holds true because each step of a path in $\check{G}$ makes reasonable progress. Suppose b is in a cone C_a, where C_a is one of the cones with apex at a by which $\check{G}$ is defined. Suppose there is an edge in $\check{G}$ from a to $c \in C_a$, so that c is a closest point in $S \cap C_a$ to a. Then if the goal is to travel from a

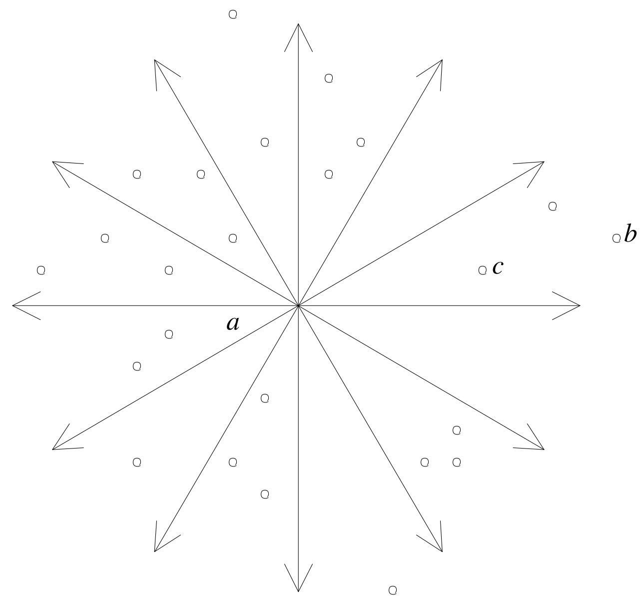

Figure 1: Cones for edges of $\check{G}$ incident to a

to b, moving from a to c makes some progress: the distance to b is reduced by an amount proportional to the distance between a and c. If the remainder of a path from a to b satisfies the same condition, then overall the length of a path $a \overset{\check{G}}{\longrightarrow} b$ will be within a small factor of the distance between a and b. The same also holds for the shortest path in $\check{G}$ between a and b.

Of course, this result is not very interesting, since the shortest path between two points is a straight line, in the absence of obstacles. However, a similar fact holds true even if obstacles are present. To see this, first note that it suffices to show that the result holds for pairs of vertices that are visible to each other, since a path in $\mathcal{V}$ corresponds to a sequence of such pairs. If an edge $a \overset{\mathcal{V}}{-} b$ exists for two vertices a and b, so that a and b are visible to each other, then consider the edge $a \overset{\mathcal{V}_\epsilon}{-} c$ such that $c, b \in C_a$ for some cone C_a with apex a. As for the obstacle-free case, movement from a to c makes proportional progress toward b. With obstacles present, this is not a sufficient condition for showing the existence of a short path $a \overset{\mathcal{V}_\epsilon}{\longrightarrow} b$. For example, in Figure 2, moving from a to c' makes no real progress toward b, since an obstacle forces motion back to a if a relatively short path to b is to be obtained. Thus, it is critical that c is a visible obstacle point that is closest to a in C_a, as this implies that no such obstacle can be present, and no "garden paths" are taken. This condition ensures the result desired.

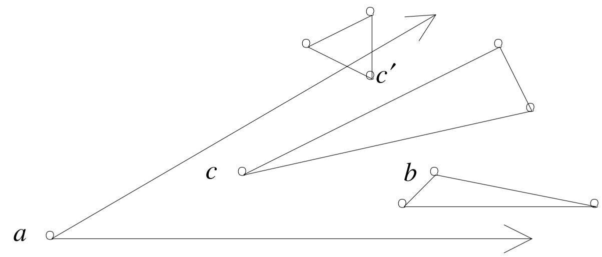

Figure 2: Edge $a \xrightarrow{\mathcal{V}_\epsilon} c$ has c closest to a in C_a.

To obtain ϵ-short paths, the above construction is modified by using $\Theta(1/\epsilon)$ cones, each with an angular span proportional to ϵ.

In the three-dimensional case, an analogous technique is used, complicated by the fact that a shortest path need not pass through vertices of the obstacles. It must, however, pass through edges of the obstacles [SS84], so that the nodes (vertices) of the appropriate visibility graph $\mathcal{V}$ in E^3 are all the points on the edges of S. In the algorithm given here, an analog of the cone construction for E^2 is (conceptually) applied to every node of the visibility graph, yielding a subgraph $\mathcal{V}_\epsilon$ with the same nodes but with a small number of graph edges per node. (Actually, to simplify the computation of $\mathcal{V}_\epsilon$, some additional nodes are included in it.) A shortest path in $\mathcal{V}_\epsilon$ is an ϵ-short path, as in the planar case.

Although $\mathcal{V}_\epsilon$ in E^3 has uncountably many nodes, its connectivity relations can be represented finitely due to the piecewise linearity of the distance function between obstacle edges. (For each cone C, this distance function is a linear approximation to the Euclidean distance from a point a to a point in C_a.) The result is that $\mathcal{V}_\epsilon$ paths between obstacle edges are combinatorially equivalent within certain intervals on the those edges. (The number of such intervals on an edge of S is $O(n\lambda(n))/\epsilon^2$, where $\lambda(n) = O(\alpha(n)^{O(\alpha(n)^{O(1)})})$ is an extremely slowly growing function, arising in the study of Davenport–Schinzel sequences [SCK$^+$86].) The preprocessing phase of the algorithm determines the set K_ϵ of these intervals, and the appropriate information for each interval to represent $\mathcal{V}_\epsilon$.

The shortest path problem on $\mathcal{V}_\epsilon$ is still not sufficiently discretized to allow computation. To do this, the approach like that of [Pap85] is followed, and the edges of S are split up into a set P of small line segments. The endpoints of these segments form some of the vertices of a subgraph $\mathcal{V}_\epsilon/P$ of $\mathcal{V}_\epsilon$, such that a short path from a point on an edge can be found via an endpoint of the segment of P containing that point. A discrete shortest path algorithm can then be applied to $\mathcal{V}_\epsilon/P$.

The approximation of $\mathcal{V}_\epsilon$ using small segments is done when a pair of source and destination points is given. Two such approximations, using sets P_1 and P_2, are done: first, a (1/2)-short path is found, using an approximation in which the number of segments $|P_1|$ is a function of ρ. Next, given that the length of the shortest path is roughly known, an approximation is done in which the size of the line segments in P_2 is a function of that shortest path length. This two-phase computation allows a complexity bound that depends on the sum of $\log \rho$ and $1/\epsilon$ and not on their product, in contrast with the algorithm of [Pap85].

1.4 Preliminaries

Before proving results about a subgraph $\mathcal{V}_\epsilon$ of the visibility graph $\mathcal{V}$, some preliminary definitions and results are necessary.

Obstacles. First a word about the nature of the obstacles: It is assumed that the obstacle set S forms a closed polyhedral region. Such a region is a subset of E^d whose boundary is the union of a collection of convex polytopes. These polytopes form a complex: that is, the intersection of two polytopes P and Q in the collection is either empty, or another polytope in the collection that is a face of P and Q. Paths from point to point must be contained in the complement of the interior of the obstacle region.

Some geometric notation. The following geometric notation will be useful, and is collected here for reference:

$\operatorname{vert} S$ denotes the set of vertices of the obstacle set S;

$\operatorname{edge} S$ denotes the set of edges of the obstacle set S;

$\overline{ab}$ denotes the closed line segment from point a to point b;

Δabc denotes the closed triangular region with vertices a, b, and c;

$D_{a,b}$ denotes the Euclidean distance from point a to point b;

$\operatorname{diam} R$ denotes the diameter of $R \subset E^d$, that is, $\sup_{a,b \in R} D_{a,b}$;

$\operatorname{int} R$ denotes the interior of $R \subset E^d$;

$\operatorname{bd} R$ denotes the boundary of $R \subset E^d$;

Graphs. The graphs considered here are undirected and weighted. An edge between vertices x and y of a graph G will be denoted by $x \overset{G}{-} y$, and a path between x and y will be denoted by $x \overset{G}{\longrightarrow} y$. This notation will also be used as a predicate, so that $x \overset{G}{\longrightarrow} y$ asserts that there is a path between x and y in G. The vertices of a graph may also be called nodes, to avoid confusion with the vertices of polyhedra.

The weight, or length, of an edge $x \overset{\mathcal{V}}{-} y$ will be denoted by $\ell(x \overset{\mathcal{V}}{-} y)$, and is equal to $D_{x,y}$. The length of path $x \overset{\mathcal{V}}{\longrightarrow} y$, or $x = x_0 \overset{\mathcal{V}}{-} x_1 \overset{\mathcal{V}}{-} x_2 \overset{\mathcal{V}}{-} \cdots \overset{\mathcal{V}}{-}$

$x_m = y$, is

$$\ell(x \stackrel{\mathcal{V}}{\longrightarrow} y) = \sum_{0 \le j < m} \ell(x_j \stackrel{\mathcal{V}}{\longrightarrow} x_{j+1}).$$

Cones. A *cone* $C_a \subset E^d$ with apex a is a closed, convex polyhedral set with the property that if a point b is contained in C_a, then so is the ray from a passing through b. If C is a cone and a is a point, the cone C_a will be the translation of C that has a as its apex.

The *angular diameter* of a cone C_a, denoted $\operatorname{adiam} C_a$, is defined by

$$\operatorname{adiam} C_a = \sup_{x,y \in C_a} \arccos \frac{(x-a)\cdot(y-a)}{\|x-a\|\|y-a\|}.$$

The following useful property of cones was noticed by Yao[Yao82]:

Lemma 1.1. If $C_a \subset E^d$ is a cone with $\pi/3 > \operatorname{adiam} C_a$, then for any $x, y \in C_a$,

$$D_{x,y} < \max\{D_{x,a}, D_{y,a}\}.$$

Proof. Omitted. Use the law of cosines with Δaxy.

Yao also proved the following result [Yao82]:

Lemma 1.2. For given dimension d and angle ψ, it is possible to construct a family $\mathcal{F}_\psi$ of cones, all of which have apexes at the origin, and having the properties that

$$E^d = \bigcup_{C \in \mathcal{F}_\psi} C,$$

for the space E^d of interest, and also

$$\psi > \operatorname{adiam} C,$$

for every $C \in \mathcal{F}_\psi$.

For $d = 2$ or 3, it is straightforward to construct such families $\mathcal{F}_\psi$ so that the number of cones in $\mathcal{F}_\psi$ is $O(1/\psi^{d-1})$ as $\psi \to 0$.

An approximate distance function. The algorithms given here will not require the evaluation of the Euclidean distance $D_{a,b}$ between points a and b, as an approximation $D^C_{a,b}$ can be used. This function is defined relative to a cone C with apex at the origin, as follows: Fix a unit vector u_C contained in C. For $a, b \in E^d$ with $b \in C_a$, define $D^C_{a,b}$ as $(b-a)\cdot u_C$. (If $b \notin C_a$, the function is undefined.) The following lemma is easily proven:

Lemma 1.3. For a cone C with apex at the origin, and for $a, b \in E^d$ with $b \in C_a$,

$$D^C_{a,b} \le D_{a,b} \le D^C_{a,b}/\cos \operatorname{adiam} C.$$

2 A subgraph for ϵ-short paths in the plane

In this section, a graph $\mathcal{V}_\epsilon$ will be defined that is a subgraph of the visibility graph $\mathcal{V}$ for a set S of obstacles. The subgraph $\mathcal{V}_\epsilon$ is defined for a given value $\epsilon > 0$, and satisfies the condition that for $a, b \in \operatorname{vert} S$ with $a \stackrel{\mathcal{V}}{\relbar\relbar\relbar} b$, there is a path $a \stackrel{\mathcal{V}_\epsilon}{\relbar\relbar\relbar} b$ in $\mathcal{V}_\epsilon$ such that

$$\ell(a \stackrel{\mathcal{V}}{\relbar\relbar\relbar} b) \leq (1+\epsilon)\ell(a \stackrel{\mathcal{V}_\epsilon}{\relbar\relbar\relbar} b).$$

As noted above, to prove this property of $\mathcal{V}_\epsilon$ for all paths in $\mathcal{V}$, it suffices to prove it for all edges of $\mathcal{V}$. Also, the number of edges in $\mathcal{V}_\epsilon$ is $O(|\operatorname{vert} S|/\epsilon)$.

The subgraph $\mathcal{V}_\epsilon$ is constructed as follows: For given $\epsilon > 0$ with $\epsilon \leq \pi$, construct a family of cones $\mathcal{F}_\psi$, where

$$\psi = \begin{cases} \min\{\pi/12, \epsilon/2\} & \text{if } \epsilon \leq \pi/2 \\ \epsilon/6 & \text{if } \pi/2 \leq \epsilon \leq \pi \end{cases}$$

For each $C \in \mathcal{F}_\psi$, and for each $a \in \operatorname{vert} S$, include $a \stackrel{\mathcal{V}_\epsilon}{\relbar} c$, where $c \in C_a \cap \operatorname{vert} S$ satisfies $a \stackrel{\mathcal{V}}{\relbar} c$, and

$$D^C_{a,c} = \min\{D^C_{a,b} \mid b \in C_a \cap \operatorname{vert} S, a \stackrel{\mathcal{V}}{\relbar} b\}.$$

In order to prove the desired results about $\mathcal{V}_\epsilon$, it will be useful to prove facts about the triangle Δbcd shown in Figure 3, where $a \stackrel{\mathcal{V}}{\relbar} b$, $a \stackrel{\mathcal{V}_\epsilon}{\relbar} c$, and d is a point on $\overline{ab}$ with $D^C_{a,c} = D^C_{a,d}$. Here $C \in \mathcal{F}_\psi$ is a cone with $b, c \in C_a$. To prove the desired result about $\mathcal{V}_\epsilon$, it will be shown that there is a path $c \stackrel{\mathcal{V}}{\relbar\relbar\relbar} b$ contained in Δbcd, such that the path $a \stackrel{\mathcal{V}_\epsilon}{\relbar} c \stackrel{\mathcal{V}}{\relbar\relbar\relbar} b$ is short. Furthermore, the path $c \stackrel{\mathcal{V}}{\relbar\relbar\relbar} b$ has the property that all edges in that path are shorter than $\ell(a \stackrel{\mathcal{V}}{\relbar} b)$. This provides the foundation for an inductive proof of the desired result about $\mathcal{V}_\epsilon$.

Lemma 2.1. For points a, b, c, d as in Figure 3, it holds that $\overline{cd} \cap \operatorname{int} S = \{\}$.

Proof. Omitted. The basic idea is to show that $\overline{cd} \cap \operatorname{int} S \neq \{\}$ implies that there is some point $c' \in \operatorname{vert} S$ contained in Δacd such that $D^C_{a,c'} < D^C_{a,c}$, and $a \stackrel{\mathcal{V}}{\relbar} c'$, contradicting the choice of c.

Proof. It will be shown that $\overline{cd} \cap \operatorname{int} S \neq \{\}$ implies that there is some point $c' \in \operatorname{vert} S$ contained in Δacd such that $D^C_{a,c'} < D^C_{a,c}$, and $a \stackrel{\mathcal{V}}{\relbar} c'$, contradicting the choice of c.

Indeed, let $c' \in S$ satisfy

$$D^C_{a,c'} = \min_{x \in S \cap \Delta acd} D^C_{a,x}.$$

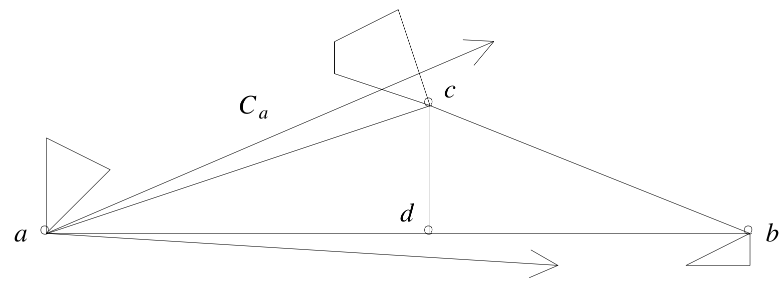

Figure 3: A path from a to b via c.
Conditions $a \stackrel{\mathcal{V}}{\text{---}} b$, $a \stackrel{\mathcal{V}_\epsilon}{\text{---}} c$, and $D^C_{a,c} = D^C_{a,d}$ hold.

Then c' is visible from a, since otherwise some point in $S \cap \Delta acd$ would be closer to a (as measured by D^C). Also $D^C_{a,c'} < D^C_{a,c}$, since $\overline{cd} \cap \operatorname{int} S \neq \{\}$ implies $(\operatorname{int} \Delta acd) \cap \operatorname{int} S \neq \{\}$, and any point of $\operatorname{int} \Delta acd$ is closer in D^C to a than c.

It remains to show that $c' \in \operatorname{vert} S$. It is easy to see that $c' \in \operatorname{bd}(\Delta acd \cap S)$, or more particularly, $c' \in \Delta acd \cap \operatorname{bd} S \subset \operatorname{bd}(\Delta acd \cap S)$. Since $\operatorname{bd} S = \cup_{e \in \operatorname{edge} S} e$, it follows that $c' \in \Delta acd \cap e$, for some $e \in \operatorname{edge} S$.

For a line segment $\overline{xy} \subset C_a$, the minimum value of $D_{a,z}$ for $z \in \overline{xy}$ is attained either at x or y (or perhaps both). Therefore c' is an endpoint of $\Delta acd \cap e$, for some $e \in \operatorname{edge} S$. It follows that either c' is an endpoint of e, or e crosses $\operatorname{bd} \Delta acd$ and $c' \in \operatorname{bd} \Delta acd$. However, $c' \notin \overline{cd}$, since $D^C_{a,c'} < D^C_{a,c}$. Furthermore, $c' \notin \overline{ac}$, since $a \stackrel{\mathcal{V}}{\text{---}} c$ implies that $\overline{ac} \cap \operatorname{int} S = \{\}$, which implies no edge of $\operatorname{edge} S$ crosses $\overline{ac}$. Similarly, no edge of $\operatorname{edge} S$ crosses $\overline{ad}$ since $a \stackrel{\mathcal{V}}{\text{---}} b$. Therefore, c' is an endpoint of e, and $c' \in \operatorname{vert} S$.

To sum up, $\overline{cd} \cap \operatorname{int} S \neq \{\}$ implies that there is some point $c' \in \Delta acd \cap \operatorname{vert} S$ with $D^C_{a,c'} < D^C_{a,c}$, and $a \stackrel{\mathcal{V}}{\text{---}} c'$. This contradicts the choice of c, so that $\overline{cd} \cap \operatorname{int} S$ must be empty. ∎

Lemma 2.2. For points a, b, c, d as in Figure 3, there is a path $c = x_0 \stackrel{\mathcal{V}}{\text{---}} x_1 \stackrel{\mathcal{V}}{\text{---}} x_2 \stackrel{\mathcal{V}}{\text{---}} \cdots \stackrel{\mathcal{V}}{\text{---}} x_m = b$ such that $x_i \in \Delta bcd$, for $0 \leq i \leq m$, and every x_i is a vertex of the convex hull of $S \cap \Delta bcd$.

Proof. Omitted.

(Note that if a, c, and b are collinear, then the lemma is trivially true, so it can be assumed that Δabc has positive area.)

Let S^* denote the the convex hull of $S \cap \Delta bcd$. Note that c and b are vertices of S^*, since they are in $S \cap \Delta bcd$ and are vertices of Δbcd. This implies that $\overline{cb}$ is an edge of S^*.

To show that $x_i \overset{\mathcal{V}}{\text{—}} x_{i+1}$, for $0 \leq i \leq m$, first it will be proven that

$$S^* \cap \operatorname{int} S \subset \overline{cb}, \tag{1}$$

and then it will be proven that

$$\operatorname{vert} S^* \subset \operatorname{vert} S. \tag{2}$$

Since S^* is the convex hull of $S \cap \Delta bcd$, the intersection $S^* \cap \operatorname{int}(S \cap \Delta bcd)$ is empty. It is shown in the next paragraph that

$$\operatorname{bd}(S \cap \Delta bcd) \subset \overline{cb} \cup \operatorname{bd} S, \tag{3}$$

so that

$$S^* \cap S = S^* \cap S \cap \Delta bcd \subset \overline{cb} \cup \operatorname{bd} S.$$

This implies (1), as desired.

It is easy to show that S and Δbcd closed implies

$$\operatorname{bd}(S \cap \Delta bcd) = (\Delta bcd \cap \operatorname{bd} S) \cup (S \cap \operatorname{bd} \Delta bcd).$$

By Lemma 2.1, $\overline{cd} \cap S \subset \operatorname{bd} S$, and since $a \overset{\mathcal{V}}{\text{—}} b$, also $\overline{bd} \cap S \subset \operatorname{bd} S$. Therefore $S \cap \operatorname{bd} \Delta bcd \subset \overline{cb} \cup \operatorname{bd} S$. Since $\Delta bcd \cap \operatorname{bd} S \subset S$, (3) follows.

Claim (2) above follows from (3), since $\operatorname{vert} S^* \subset \operatorname{vert}(S \cap \Delta bcd)$, and from (3), $\operatorname{vert}(S \cap \Delta bcd) \subset \overline{cb} \cup \operatorname{vert} S$. Since $\overline{cb} \cap \operatorname{vert} S^* = \{\}$, relation (2) follows, and so the lemma. ∎

Lemma 2.3. If A and B are closed, bounded, and convex subsets of E^d with $A \subset B$, then the surface area of A is no more than the surface area of B. Therefore, for points a, b, c, d as in Figure 3, the path $c \overset{\mathcal{V}}{\text{——}} b$ described by the previous lemma has the property that

$$\ell(c \overset{\mathcal{V}}{\text{——}} b) \leq D_{c,d} + D_{d,b}.$$

Proof. The general statement is proven in [Egg58, 5.3], from which the particular claim follows. ∎

(This lemma is a special case of the more general condition that if A and B are convex sets with $A \subset B$, then the surface area of A is no more than the surface area of B.)

For each $x_i \overset{\mathcal{V}}{\text{—}} x_{i+1}$, let y_i be the point such that $\overline{x_i y_i}$ is parallel to $\overline{cd}$, and $\overline{y_i x_{i+1}}$ is parallel to $\overline{db}$. Then by the triangle inequality,

$$\ell(x_i \overset{\mathcal{V}}{\text{—}} x_{i+1}) \leq D_{x_i,y_i} + D_{y_i,x_{i+1}},$$

so that

$$\ell(c \overset{\mathcal{V}}{\longrightarrow} b) \leq \sum_{0 \leq i < m} D_{x_i,y_i} + D_{y_i,x_{i+1}}.$$

But

$$D_{c,d} = \sum_{0 \leq i < m} D_{x_i,y_i},$$

and

$$D_{d,b} = \sum_{0 \leq i < m} D_{y_i,x_{i+1}},$$

so the lemma follows. ∎

The elements are almost in place for an inductive proof that $\mathcal{V}_\epsilon$ contains short paths. The following two lemmas provide a means of using induction on the length of the edges of $\mathcal{V}$.

Lemma 2.5. For any given points $a, b, c \in E^d$, it holds that

$$\operatorname{diam} \Delta abc = \max_{\{x,y\} \subset \{a,b,c\}} D_{x,y}.$$

Proof. Omitted.

Lemma 2.4. For points a, b, c, d as in Figure 3, and with $\psi < \pi/4$, if $x, y \in \Delta bcd$, then $D_{x,y} < D_{a,b}$.

Proof. Omitted. Note that it is enough to show that $D_{c,b}$, $D_{b,d}$, and $D_{c,d}$ are each less than $D_{a,b}$.

By Lemma 2.5**fix**, it is enough to show that $D_{c,b}$, $D_{b,d}$, and $D_{c,d}$ are each less than $D_{a,b}$.

Plainly $D_{b,d} < D_{a,b}$. The fact that $D_{c,b} < D_{a,b}$ follows from

$$\begin{aligned} D^2_{c,b} &= D^2_{a,b} + D^2_{a,c} - 2D_{a,c}D_{a,b}\cos\theta \\ &= D^2_{a,b} + D_{a,c}(D_{a,c} - 2D_{a,b}\cos\theta), \end{aligned}$$

where $\theta \leq \psi$ is the measure of $\angle cab$. By Lemma 1.3,

$$D_{a,c} \leq D_{a,b}/\cos \operatorname{adiam} C = D_{a,b}/\cos\psi,$$

so $D_{a,c} < 2D_{a,b}\cos\theta$ if

$$D_{a,b}/\cos\psi < 2D_{a,b}\cos\psi < 2D_{a,b}\cos\theta.$$

The fact that $D_{c,b} < D_{a,b}$ follows, since $\psi < \pi/4$ implies $\cos^2\psi > 1/2$.

A similar argument shows that $D_{c,d} < D_{a,d}$, which implies $D_{c,d} < D_{a,b}$. ∎

Theorem 2.5. Let S be a set of obstacles, $\mathcal{V}$ a vertex visibility graph, and $\mathcal{V}_\epsilon$ the subgraph of $\mathcal{V}$ defined above. Then for any points $a, b \in \operatorname{vert} S$, if $a \overset{\mathcal{V}}{\relbar\relbar} b$ then $a \overset{\mathcal{V}_\epsilon}{\relbar\relbar} b$, and

$$\ell(a \overset{\mathcal{V}_\epsilon}{\relbar\relbar} b) \leq (1+\epsilon)\ell(a \overset{\mathcal{V}}{\relbar\relbar} b).$$

Proof. As noted previously, it is enough to show that the above conditions hold when the path $a \overset{\mathcal{V}}{\relbar\relbar} b$ consists only of a single edge.

The proof will proceed by induction, using the ordering on the edges of $\mathcal{V}$ implied by their length.

For the inductive basis, suppose $a \overset{\mathcal{V}}{\relbar} b$ satisfies

$$\ell(a \overset{\mathcal{V}}{\relbar} b) \leq \min_{x,y \in \operatorname{vert} S} \ell(x \overset{\mathcal{V}}{\relbar} y).$$

If $a \overset{\mathcal{V}_\epsilon}{\relbar} b$ does *not* hold, then there is some c with $a \overset{\mathcal{V}_\epsilon}{\relbar} c$, hence by Lemmas 2.3 and 2.4, there is some $c' \in \Delta bcd$ with $c' \overset{\mathcal{V}}{\relbar} b$ and $\ell(c' \overset{\mathcal{V}}{\relbar} b) < \ell(a \overset{\mathcal{V}}{\relbar} b)$. This contradicts the assumption about $a \overset{\mathcal{V}}{\relbar} b$, so that $a \overset{\mathcal{V}_\epsilon}{\relbar} b$.

Now suppose $a \overset{\mathcal{V}}{\relbar} b$ is an arbitrary edge of $\mathcal{V}$, and by the inductive hypothesis, the claim holds for all edges of $\mathcal{V}$ that are shorter than $a \overset{\mathcal{V}}{\relbar} b$. There is a point c with $a \overset{\mathcal{V}_\epsilon}{\relbar} c$, and with $c, b \in C_a$, for some cone $C \in \mathcal{F}_\psi$. By Lemmas 2.2 and 2.4, there is a path $c \overset{\mathcal{V}}{\relbar\relbar} b$ with all edges in that path shorter than $\ell(a \overset{\mathcal{V}}{\relbar} b)$. By the inductive hypothesis and Lemma 2.3, there is a path $b \overset{\mathcal{V}_\epsilon}{\relbar\relbar} c$ such that

$$\ell(c \overset{\mathcal{V}_\epsilon}{\relbar\relbar} b) < (1+\epsilon)\ell(c \overset{\mathcal{V}}{\relbar\relbar} b) \leq (1+\epsilon)(D_{c,d} + D_{d,b}).$$

This implies

$$\ell(a \overset{\mathcal{V}_\epsilon}{\relbar} c \overset{\mathcal{V}_\epsilon}{\relbar\relbar} b) < D_{a,c} + (1+\epsilon)(D_{c,d} + D_{d,b}).$$

The theorem follows from

$$D_{a,c} + (1+\epsilon)(D_{c,d} + D_{d,b}) \leq (1+\epsilon)D_{a,b}, \qquad (?)$$

which readily follows from the relation

$$1 + (1+\epsilon)(\sin\psi - \cos\psi) \leq 0.$$

∎

To prove the theorem, it remains to show that

$$D_{a,c} + (1+\epsilon)(D_{c,d} + D_{d,b}) \leq (1+\epsilon)D_{a,b}. \qquad (1)$$

Let θ_a denote the angle $\angle dac$ at a in Δacd, and similarly let θ_c and θ_d denote the angles at c and d in Δacd. Then by the law of sines, $D_{a,c} = D_{a,d}\sin\theta_d/\sin\theta_c$ and $D_{c,d} = D_{a,d}\sin\theta_a/\sin\theta_c$, so (1) is equivalent to

$$D_{a,d}\frac{\sin\theta_d}{\sin\theta_c} + (1+\epsilon)(D_{a,d}\frac{\sin\theta_a}{\sin\theta_c} + D_{d,b}) \leq (1+\epsilon)D_{a,b},$$

or

$$\sin\theta_d + (1+\epsilon)(\sin\theta_a - \sin\theta_c) \leq 0. \tag{2}$$

The fact that a, b, c, and $a + u_C$ are all in C_a implies that $\theta_a \leq \psi$ and $\theta_c \geq \pi/2 - \psi$, so that the left hand side of (2) is bounded above by

$$1 + (1+\epsilon)(\sin\psi - \sin(\pi/2-\psi)) = 1 + (1+\epsilon)(\sin\psi - \cos\psi).$$

It is easy to verify that the relationship between ϵ and ψ in the definition of $\mathcal{V}_\epsilon$ ensures that this value is less than zero. This implies that (1) holds, and the inductive step is accomplished. ∎

3 Finding ϵ-short paths in the plane

In the previous section, a subgraph $\mathcal{V}_\epsilon$ was given, satisfying the condition that $\mathcal{V}_\epsilon$ contains ϵ-short paths between any two vertices of an obstacle set S. With such a subgraph available, an ϵ-short path between any points s and t can be found readily: Given s and t, augment $\mathcal{V}_\epsilon$ with s and t as two more vertices, and include for s and t edges analogous to those in $\mathcal{V}_\epsilon$. That is, for s, and for every $C \in \mathcal{F}_\psi$, find the vertex x of $\mathcal{V}_\epsilon$ in C_s that is visible to s, and is closest among all such visible vertices in C_s. Make $s \xrightarrow{\mathcal{V}_\epsilon} x$. Perform the analogous operation for t and every $C \in \mathcal{F}_\psi$. Then Theorem 2.5 holds for the graph $\mathcal{V}_\epsilon$ augmented in this way, and a shortest path in $\mathcal{V}_\epsilon$ from s to t is an ϵ-short path from s to t. Using the algorithm of Fredman and Tarjan[FT84], such a path can be found in $O(n\log n + n/\epsilon)$ time.

How can such an augmented subgraph $\mathcal{V}_\epsilon$ be found quickly? It is shown in this section that this can be done using *conical Voronoi diagrams*, or C-VoDs. Given an obstacle set S and a cone C, the C-VoD for C and S is a set of regions, with a region V_x for every $x \in \operatorname{vert} S$. For any point $y \in V_x$, a D^C-closest point to y in C_y is x. More formally, the region V_x is defined as follows: Let V_x^* denote the set of points in $(-C)_x \setminus \{x\}$ that are visible to x; that is, V_x^* is the set of points y with $y \neq x$ and with $x \in C_y$. For $y \in E^2$, let

$$D_y^* = \min\{D_{y,x'}^C \mid x' \in \operatorname{vert} S, y \in V_{x'}^*\},$$

and

$$V_x^{**} = \{y \in V_x^* \mid D_{y,x} = D_y^*\}.$$

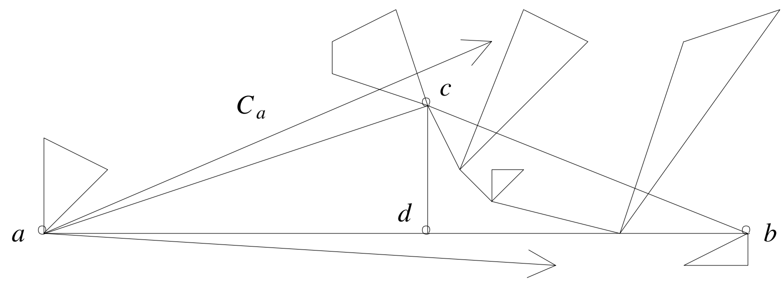

Figure 4: Some regions of a C-VoD.

Then to break ties, V_x is defined as the set of points $y \in V_x^{**}$ such that if $y \in V_{x'}^{**}$, then x is to the right of x' when facing in the direction of u_C. This convention makes the Voronoi regions disjoint. Note that in this definition $x \notin V_x$. See Figure 4 for an example.

It is easy to see that for $x \in \operatorname{vert} S$, the edge $x \xrightarrow{\mathcal{V}_\epsilon} y$ can be included if $y \in V_x$, where either $y \in \operatorname{vert} S$ or y is a source or destination point. It is shown in the full paper that a C-VoD is a polygonal subdivision with $O(n)$ edges, a C-VoD can be constructed in $O(n \log n)$ time using a sweepline algorithm, and the C-VoD region containing each vertex is determined as a by-product of that algorithm. Thus the construction of the C-VoDs for every $C \in \mathcal{F}_\psi$ will allow the construction of a subgraph $\mathcal{V}_\epsilon$ in $O(n \log n) O(1/\epsilon)$ time. Augmentation of $\mathcal{V}_\epsilon$ for given source point a and destination point b can be done in $O(n/\epsilon)$ time, within the bound required for determining a shortest path tree.

In the remainder of this section, the problem considered is that of computing a C-VoD for an obstacle set S and a fixed cone C.

It will be convenient to use a coordinate system (x, y) so that $u_C = (0, 1)$. For a point $a \in E^2$, a_x will denote the x coordinate of a, and a_y the y coordinate. For $a, b \in E^2$, the partial order $a \prec b$ will be true just when either $a_y > b_y$, or $a_y = b_y$ and $a_x < a_y$. That is, either $u_C \cdot a_y > u_C \cdot b_y$, or $u_C \cdot a_y = u_C \cdot b_y$ and b is to the right of a when facing in the direction of u_C. The relation $a \succ b$ is defined in the obvious way as $b \prec a$, and $a \preceq b$ means that either $a \prec b$ or $a = b$. The relation $\succeq$ is similarly defined.

Here are some basic properties of C-VoDs, using these definitions.

Lemma 3.1. For a cone C and a polyhedral obstacle set S consisting of two points a and b with $a \prec b$, the C-VoD region $V_b = V_b^*$, and $V_a = V_a^* \setminus V_b^*$.

Proof. For $c \in E^2$ with $a, b \in C_c \setminus \{c\}$, the distance $D_{y,a}^C = (a - y) \cdot u_C$, and $D_{y,b}^C = (b - y) \cdot u_C$, so either $D_{y,a}^C > D_{y,b}^C$, or $D_{y,a}^C = D_{y,b}^C$, and b is to the right of a. In either case, $y \in V_b$ by the definition of V_a. Thus, $c \in V_a$ only when $c \in V_a^*$ but $c \notin V_b^*$. The lemma follows. ∎

The following generalization of Lemma 3.1 follows readily.

Lemma 3.2. For a polyhedral obstacle set S and a point $a \in \operatorname{vert} S$, the C-VoD region

$$V_a = V_a^* \setminus \bigcup_{\substack{b \in \operatorname{vert} S \\ a \prec b}} V_b^*.$$

Proof. Omitted.

From Lemma 3.2 follows:

Lemma 3.3. For $a \in \operatorname{vert} S$,

$$\operatorname{bd} V_a \subset \operatorname{bd} V_a^* \cup \bigcup_{\substack{b \in \operatorname{vert} S \\ a \prec b}} \operatorname{bd} V_b^*.$$

Proof. Omitted.

To characterize the boundary of V_a, the following fact about the boundary of V_a^* will be useful.

Lemma 3.4. For $a \in \operatorname{vert} S$, either a point $b \in \operatorname{bd} V_a^*$ is in $\operatorname{bd}(-C)_a \cup \operatorname{bd} S$, or there is some $c \in \operatorname{vert} S$ with a, c, and b on a line, c between a and b.

Proof. Omitted.

Lemma 3.5. For $a \in \operatorname{vert} S$,

$$\operatorname{bd} V_a \subset \operatorname{bd} S \cup \bigcup_{\substack{f \in \operatorname{vert} S \\ f \succeq a}} \operatorname{bd}(-C)_f.$$

By Lemma 3.3, if $b \in \operatorname{bd} V_a$, then $b \in \operatorname{bd} V_c^*$, for some $c \succeq a$. Suppose $b \notin \operatorname{bd}(-C)_c \cup \operatorname{bd} S$. Then by Lemma 3.4, there is some $d \in \operatorname{vert} S$ with a, d, and b on a line, d between a and c. Let d^* be the "lowest" such d, the one closest to b. Plainly $d^* \succeq a$, and either $b \in \operatorname{int} V_d^*$ or $b \in \operatorname{bd}(-C)_d \cup \operatorname{bd} S$. In the former case, b and a neighborhood of b are not in V_a, by Lemma 3.2, and so $b \notin \operatorname{bd} V_a$, contradiction. In the latter case, b is in the set indicated in the lemma statement. ∎

The above lemma, and the next one, imply that $\operatorname{bd} V_a$ has a simple structure.

Lemma 3.6. For $a \in \operatorname{vert} S$, V_a is starshaped from a. That is, if $b \in V_a$, then $\overline{ba} \subset V_a \cup \{a\}$.

Proof. Suppose $b \in V_a$. Then any point in $\overline{ba}$ is visible to a. Suppose $c \in \overline{ba}$, but there is some $d \in \operatorname{vert} S$ with $c \in V_d^*$, and $d \succ a$. Then d cannot be visible to b, since that would contradict $b \in V_a$. By a proof like that for Lemma 2.1, these conditions imply that some $f \in \operatorname{vert} S$ in Δbcd is visible to b. But by the assumptions about c and d, we have $f \in (-C)_a$ and $f \succ a$, which imply $b \notin V_a$, contradiction. ∎ As shown in the lemma below, the boundary (of the closure) of a C-VoD region V_a is the union of a set of line segments and rays. Let a *cone segment* e be a ray or line segment that has an endpoint at some $a \in \operatorname{vert} S$, and satisfies $e \subset \operatorname{bd} C_a$. Let an *edge segment* be a line segment that is a subset of some edge in $\operatorname{edge} S$.

Theorem 3.7. For $a \in \operatorname{vert} S$, the boundary of the closure of V_a is the union of cone segments and edge segments.

Proof. The lemma follows immediately from Lemmas 3.5 and 3.6. ∎

Theorem 3.8. For an obstacle set S with n edges, the total number of edges in a C-VoD of S is $O(n)$.

Proof. Since there are $O(n)$ vertices of S, and at most 2 cone edges per vertex, there are $O(n)$ cone edges. Each edge segment is either a whole edge, or is in some V_a, and has one endpoint in $\operatorname{bd}(-C)_a$. Thus there are $O(n)$ edge segments, and so by Theorem 3.7, there are $O(n)$ edges. ∎

Theorem 3.9. For an obstacle set S with n edges, a C-VoD of S can be computed by a sweepline algorithm in $O(n \log n)$ time.

Proof. Omitted.

Sweepline algorithms are well known. For this problem, a sweepline algorithm somewhat like Fortune's [For86] can be used. Fortune's algorithm uses a geometric transformation, so that the transform of the Voronoi region for a site is not encountered in the sweep until the site is. This transformation is not necessary in this case, because by sweeping in the appropriate direction, the Voronoi region for a vertex is encountered when that vertex is swept over.

4 A graph for ϵ-short paths in E^3

In this section and the next, an algorithm is described for finding ϵ-short paths through polyhedral obstacles in E^3. As described in §1, the algorithm combines the ideas of the last section with those of Papadimitriou [Pap85].

The graph $\mathcal{V}$ in E^3 is defined analogously to the planar case, except that in E^3, the graph's nodes are all the points of $\operatorname{edge} S$. The graph $\mathcal{V}_\epsilon$ is also defined analogously to the planar case. A family $\mathcal{F}_\psi$ of cones is constructed with angular diameter $\psi = \Omega(\epsilon)$, so that the number of cones in $\mathcal{F}_\psi$ is $\Theta(1/\epsilon^2)$. It will be assumed that the cones in $\mathcal{F}_\psi$ have three sides. The nodes of $\mathcal{V}_\epsilon$ are of two types, the *ordinary* nodes, and the *Steiner* nodes. The set of ordinary nodes corresponds to the nodes of $\mathcal{V}$, that is, the set of points on edges of S. The set of Steiner nodes results from the following procedure for constructing $\mathcal{V}_\epsilon$: For each ordinary node a and each $C \in \mathcal{F}_\psi$, find the point b in $S \cap C_a$ that is D^C-closest to a. (If $a \in e \in \operatorname{edge} S$ and $e \cap C_a \neq \{\}$, include no $\mathcal{V}_\epsilon$ edge for a and C.)

Note that b need not be not an obstacle edge, that is, b is not necessarily an ordinary node. If b is not an ordinary node, make b a Steiner node. In either case, include the edge $a \overset{\mathcal{V}_\epsilon}{\text{---}} b$.

With the inclusion of graph nodes in the interiors of the obstacle facets, it is necessary to include in $\mathcal{V}_\epsilon$ edges among the nodes on each obstacle facet. Such nodes are the Steiner nodes, together with the ordinary nodes on the edges of the facet. The edges can be defined as in the planar case, using a cone family $\mathcal{F}_{\psi'}$, such that between any two nodes on a facet, there is a path in $\mathcal{V}_\epsilon$ that is ϵ'-short, for some $\epsilon' > 0$. It will be useful to pick a value of ϵ' that is less than ϵ, although $\epsilon' = \Theta(\epsilon)$ will hold.

The advantage of defining $\mathcal{V}_\epsilon$ with the use of Steiner nodes is that the computation of $\mathcal{V}_\epsilon$ becomes easier, due to the simpler criterion for determining the $\mathcal{V}_\epsilon$ edges. The disadvantage of this definition is that the proof that $\mathcal{V}_\epsilon$ contains ϵ-short paths becomes harder. The next lemma gives the desired condition.

Lemma 4.1. Let S be a set of obstacles in E^3, $\mathcal{V}$ the visibility graph, and $\mathcal{V}_\epsilon$ the graph defined above. Then for any points a and b on edges of S, if $a \overset{\mathcal{V}}{\text{---}} b$ then $a \overset{\mathcal{V}_\epsilon}{\text{------}} b$, and

$$\ell(a \overset{\mathcal{V}_\epsilon}{\text{------}} b) \leq (1+\epsilon)\ell(a \overset{\mathcal{V}}{\text{---}} b).$$

The lemma says that there are ϵ-short paths between ordinary nodes that are visible to each other. To prove the lemma inductively, it will be useful to also have certain facts about paths involving Steiner nodes:

Lemma 4.2. With terminology as in Lemma 4.1, let a be an ordinary node, and let b be a point on some obstacle facet F with a visible to b and with $\overline{ab}$ perpendicular to F. Then there is some node $c \in F$ with $b, c \in C_a$ for some $C \in \mathcal{F}_\psi$, such that $a \overset{\mathcal{V}_\epsilon}{\text{------}} c$ and

$$\ell(a \overset{\mathcal{V}_\epsilon}{\text{------}} c) + D_{c,b} < (1+\epsilon)D_{a,b}.$$

Proof of Lemmas 4.1 and 4.2. (Sketch) These two lemmas will be proven jointly.

Suppose $a \overset{\mathcal{V}_\epsilon}{\text{---}} c$, where $c, b \in C_a$ for some $C \in \mathcal{F}_\psi$. Then by considering the intersection of the plane determined by a, b, and c with S, a proof similar to that for the planar case can be given. The proof is more complicated, however, because of the Steiner nodes, and because the number of nodes is infinite.

The motivation for Lemma 4.2 is the situation shown in Figure 5, where a is an ordinary node, b and c are Steiner nodes, a is visible to b, $a \overset{\mathcal{V}_\epsilon}{\text{---}} c$, and $b, c \in C_a$

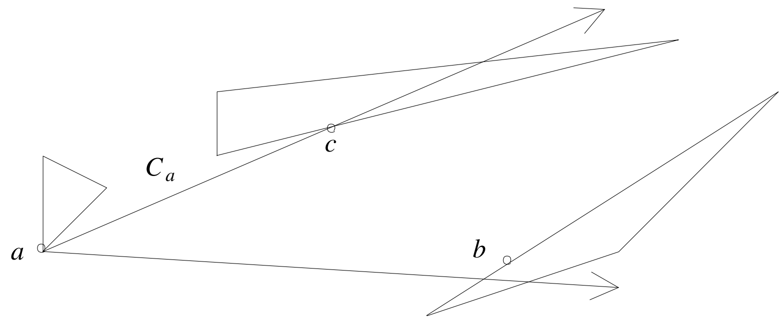

Figure 5: A bad case for Lemma 4.1 without Lemma 4.2.

for some cone C. The problem here is to find an ϵ-short path from a to b via c. This situation could arise as a subproblem in finding an ϵ-short path from a to some other ordinary node. There is no obvious means of making sure that a path exists in $\mathcal{V}_\epsilon$ between c and b without including edges in $\mathcal{V}_\epsilon$ that are between two Steiner nodes on different facets. Such a course leads to the introduction of yet more Steiner nodes, and so on. To avoid such a difficulty, Lemma 4.1 is proven in such a way that the only subproblems arising that involve Steiner nodes have the form described in Lemma 4.2. Figure 6 gives an indication of how this is done, and why a subproblem of the form of Lemma 4.2 is solvable.

Let S^* denote the intersection of S with the quadrilateral region $abce$. Let p_c denote a "point at infinity" in the direction perpendicular to $c'c''$, and p_b a a point at infinity in the direction perpendicular to $b'b''$. Then the convex hull of $S^* \cup \{p_e, p_b\}$ gives a path consisting of a segment s_c perpendicular to $c'c''$, a series of edges between ordinary nodes, and a segment perpendicular to the facet containing b. Although the segment s_c is not necessarily perpendicular to the facet containing c, a further construction gives a path to c that includes such a segment.

A note about the inductive argument of the proof: While the proof for E^2 used induction on the length of edges in $\mathcal{V}$, in E^3 the argument that the induction "grounds out" must be more delicate. Given nodes a and b, consider the $\mathcal{V}_\epsilon$ edges generated by attempting an inductive proof along the lines of that for E^2. If the set of edges generated becomes arbitrarily large, then there will be edges in that set that become arbitrarily small. This implies the existence in the set of edges of a sequence of edges $a_i \xrightarrow{\mathcal{V}_\epsilon} b_i$, where for some $e, f \in \operatorname{edge} S$, it holds that $a_i \in e$ and $b_i \in f$, and the lengths of the edges become arbitrarily small. This implies that a_i and b_i are converging to the point $p_{ef} = e \cap f$. However, since these graph edges can be "short circuited" by passing through p_{ef}, such sequences are not present in a shortest path in $\mathcal{V}_\epsilon$ from a to b, so that the process of generating edges for an inductive proof terminates either by such

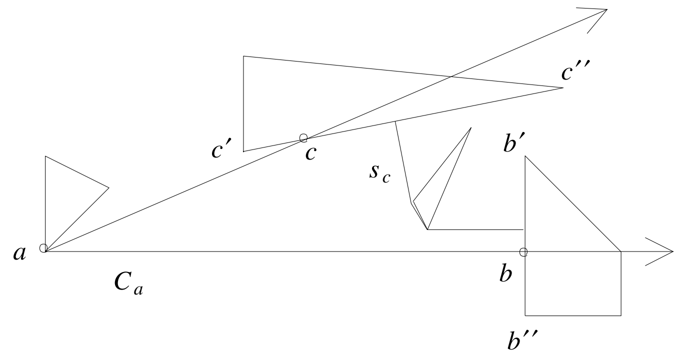

Figure 6: Proof of Lemma 4.2 by picture.

short circuiting, or by stopping with edges in $\mathcal{V}_\epsilon$. ∎

5 Finding ϵ-short paths in E^3

5.1 Combinatorial characterization of $\mathcal{V}_\epsilon$

An important concept in manipulating $\mathcal{V}_\epsilon$ is that of *combinatorial equivalence* of points in edge S. Suppose $e \in \text{edge}\, S$ has endpoints a and b. For $0 \leq \beta \leq 1$, let $p = \beta a + (1-\beta)b$. For cone $C \in \mathcal{F}_\psi$ and point p, let p_C denote the point in S so that D^C_{p,p_C} minimum over all points in S. Let $h^C(\beta)$ denote that value of D^C_{p,p_C} as a function of β. Then it is easy to see that $h^C(\beta)$ is piecewise linear, and e is naturally divided into segments within which this function is linear. Two points in e will be said to be *C-combinatorially equivalent* if they are in the same such segment of e. Two points in e will be said to be *combinatorially equivalent* if they are C-combinatorially equivalent for all $C \in \mathcal{F}_\psi$. The set of line segments forming the equivalence classes for this relation over all obstacle edges will be called the *combinatorial characterization* of $\mathcal{V}_\epsilon$, and will be denoted K_ϵ. Given K_ϵ and associated labels for its segments, it is possible to quickly determine for any $a \in e \in \text{edge}\, S$ the $\mathcal{V}_\epsilon$ edges from a.

For each facet F of S, let $h^C_F(\beta)$ denote the value of D^C_{p,p_F} as a function of β, where p_F is the closest point to p in $C_p \cap F$. Note that the function $h^C(\beta)$ takes the minimum value at β of all of the functions $h^C_F(\beta)$, That is, $h^C(\beta)$ is the "lower envelope" of these functions. Each function $h^C_F(\beta)$ is piecewise linear, with $O(1)$ pieces. This implies that $h^C(\beta)$ corresponds to a Davenport–Schinzel sequence. Such sequences are discussed in [SCK$^+$86], where an analysis is given implying that $h^C(\beta)$ has a number of pieces that is almost linear in the number of facets F. Since these pieces correspond to the blocks of C-combinatorial equivalence, we have specifically:

Lemma 5.1. The number of segments in a partition of an edge of S by C-combinatorial equivalence is bounded by $n\lambda(n)$, where $\lambda(n) = O(\alpha(n)^{O(\alpha(n)^{O(1)})})$, and $\alpha(n)$ is a form of inverse of Ackermann's function.

Theorem 5.2. For a given obstacle set $S \subset E^3$, the combinatorial characterization K_ϵ of $\mathcal{V}_\epsilon$ has $O(n^2\lambda(n)/\epsilon^2)$ segments, and can be computed in $O(n^2\lambda(n)\log n)/\epsilon^2$ time.

Proof. Omitted. The characterization is determined for each edge and each $C \in \mathcal{F}_\psi$ in turn, using a merging procedure to determine the associated lower envelope.

5.2 Finding ϵ-short paths in $\mathcal{V}_\epsilon$

Having performed the preprocessing step of finding the line segment set K_ϵ, which is a combinatorial characterization of $\mathcal{V}_\epsilon$, it remains to show how to compute an ϵ-short path between two given points s and t. As described in the introduction, this involves two phases, which use partitions P_1 and P_2 of edge S, and corresponding subgraphs $\mathcal{V}_\epsilon/P_i$ of $\mathcal{V}_\epsilon$. A segment in P_i ($i = 1$ or 2) will be called a P_i*-segment*, and a segment in K_ϵ will be called a K_ϵ-segment. The length of a segment f will be denoted by $\ell(f)$.

Given K_ϵ and a partition P_i, the nodes and edges of $\mathcal{V}_\epsilon/P_i$ are defined as follows: The nodes of $\mathcal{V}_\epsilon/P_i$ include the endpoints of segments in K_ϵ and P_i. If a is the endpoint of a segment in P_i, include in $\mathcal{V}_\epsilon/P_i$ all edges $a \xrightarrow{\mathcal{V}_\epsilon} b$. If a is the endpoint of a segment in K_ϵ, where a is the endpoint of some block of C-combinatorially equivalent edges, include in $\mathcal{V}_\epsilon/P_i$ the corresponding edge $a \xrightarrow{\mathcal{V}_\epsilon} b$, where $b \in C_a$. Include in the nodes of $\mathcal{V}_\epsilon/P_i$ all nodes incident to the above edges. Finally, include a graph edge between nodes that are adjacent on an obstacle edge, and compute a C-VoD for the nodes of $\mathcal{V}_\epsilon/P_i$ on each obstacle facet, using the family of cones $\mathcal{F}_{\psi'}$ mentioned above.

The justification for this construction is the following lemma:

Lemma 5.3. Suppose segment $f \subset e \in \operatorname{edge} S$, with the points of f C-combinatorially equivalent for some C. Then there is an endpoint a^* of f for which the following holds. Let $a^* \xrightarrow{\mathcal{V}_\epsilon} b^*$ with $b^* \in C_{a^*}$. Then for any $a \in f$ and edge $a \xrightarrow{\mathcal{V}_\epsilon} b$ with $b \in C_a$,

$$D_{a,a^*} + \ell(a^* \xrightarrow{\mathcal{V}_\epsilon} b^*) + D_{b^*,b} \leq \ell(a \xrightarrow{\mathcal{V}_\epsilon} b) + 2\ell(f).$$

In other words, approximating edges $a \xrightarrow{\mathcal{V}_\epsilon} b$ using $a^* \xrightarrow{\mathcal{V}_\epsilon} b^*$ results in an additive error proportional to the length of f.

Proof. Omitted.

Lemma 5.4. The number of nodes in $\mathcal{V}_\epsilon/P_i$ is at most $|K_\epsilon|/\epsilon + |P_i|/\epsilon^2$ and the number of edges is at most $|K_\epsilon|/\epsilon^2 + |P_i|/\epsilon^3$.

Proof. Omitted.

It remains to define the collections of segments P_1 and P_2. The set P_1 is used with $K_{1/4}$ to determine the length of a (1/2)-shortest path between two points s and t. Roughly following [Pap85], a segment $e \in \operatorname{edge} S$ is partitioned for P_1 as follows: the section of e consisting of points within $D_{s,t}$ of s is partitioned by segments of length $\epsilon_1 = 1/8n$. The other section(s) e^* of e is partitioned as follows: choose a coordinate system on e^* with the closest point to s as the origin. Divide e^* with the sequence of points

$$x_j = \epsilon_1 D_{s,t}(1+\epsilon_1)^{j-1},$$

for $j = 1, 2, \ldots$. Then $|P_1| = n^2 \log n\rho$, and the additive error for passing through an ordinary node of $\mathcal{V}_\epsilon/P_1$ is no more than $1/8n$ times the length of the shortest possible path from s to t that passes through that node. Observe that a shortest path from s to t using $\mathcal{V}_\epsilon/P_1$ passes through $2n$ ordinary nodes, 2 nodes per edge. The result is that by finding the shortest path in $\mathcal{V}_\epsilon/P_1$ augmented for s and t, the length Q of the shortest path from s to t is known with relative error at most 1/2.

Given this estimate Q for the shortest path length, the collection P_2 is defined as follows: For the set of points on an edge e that are within $2Q$ of s, partition that set into segments of length $\epsilon Q/2n$. The shortest path on $\mathcal{V}_\epsilon/P_2$ is an ϵ-short path. We have:

Theorem 5.5. Given a set S of polyhedral obstacles in E^3, and points s and t, an ϵ-short path between s and t can be found in

$$O(n^2\lambda(n)\log(n/\epsilon)/\epsilon^4 + n^2 \log n\rho \log(n \log \rho))$$

where n is the number of obstacle faces, and ρ is the ratio of the length of the longest edge in S to the distance between s and t.

Proof. The theorem follows from the previous lemmas, using the bounds for $|K_\epsilon|$, $|P_i|$, and the corresponding number of edges in $\mathcal{V}_\epsilon/P_i$. The algorithm of [FT84] requires $O(m + n\log n)$ time for the single-source shortest path problem on a graph with m edges and n vertices.

6 Generalization to l_p norms

By varying numerical parameters slightly, the algorithms and data structures of this paper can be applied to finding ϵ-short paths under arbitrary l_p norms. By suitable choice of a factor γ_p, the quantity $\gamma_p D^C_{a,b}$ is a good estimate of the l_p-distance between a and $b \in C_a$.

References

[AAG+85] Takao Asano, Tetsuo Asano, L. Guibas, J. Hershberger, and H. Imai. Visibility-polygon search and Euclidean shortest paths. In *Proc. 26th IEEE FOCS*, pages 155–164, 1985.

[Che86] P. Chew. There is a planar graph almost as good as the complete graph. In *Proc. 2nd Symp. on Comp. Geometry*, pages 169–177, 1986.

[Egg58] H. G. Eggleston. *Convexity*. Cambridge University Press, Cambridge, 1958.

[For86] S. J. Fortune. A sweepline algorithm for Voronoi diagrams. In *Proc. 2nd Symp. on Comp. Geometry*, pages 313–322, 1986.

[FT84] M. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization problems. In *Proc. 25th IEEE FOCS*, pages 338–346, 1984.

[Mou84] D. Mount. On finding shortest paths on convex polyhedra. Technical report, Department of Computer Science, Univ. of Maryland, 1984.

[Pap85] C. H. Papadimitriou. An algorithm for shortest-path motion in three dimensions. *Information Processing Letters*, 20:259–263, 1985.

[RS85] J. Reif and J. A. Storer. Shortest paths in Euclidean space with polyhedral obstacles. Technical Report CS-85-121, Department of Computer Science, Brandeis Univ., 1985.

[SB86] M. Sharir and A. Baltsan. On shortest paths amidst convex polyhedra. In *Proc. 2nd Symp. on Comp. Geometry*, pages 193–206, 1986.

[SCK+86] M. Sharir, R. Cole, K. Kedem, D. Leven, R. Pollack, and S. Sifrony. Geometric applications of Davenport-Schinzel sequences. In *Proc. 27th IEEE FOCS*, pages 77–86, 1986.

[SS84] M. Sharir and A. Schorr. On shortest paths in polyhedral spaces. In *Proc. 16th Annual SIGACT Symp.*, pages 144–153, 1984.

[Yao82] A. C. Yao. On constructing minimum spanning trees in k-dimensional spaces and related problems. *SIAM Journal on Computing*, 11:721–736, 1982.

[Yap85] C. K. Yap. Algorithmic motion planning. In J. T. Schwartz and C. K. Yap, editors, *Advance in Robotics, Volume I*. Lawrence Erlbaum Associates, 1985.