Approximate Shortest Paths and Geodesic Diameter on a Convex Polytope in Three Dimensions*

Sariel Har-Peled†

November 27, 1996

Abstract

Given a convex polytope P with n edges in $\mathbb{R}^3$, we present a relatively simple algorithm that preprocesses P in $O(n)$ time, such that, given any two points $s, t \in \partial P$, and a parameter $0 < \varepsilon \leq 1$, it computes, in $O((\log n)/\varepsilon^{1.5} + 1/\varepsilon^3)$ time, a distance $\Delta_P(s,t)$, such that $d_P(s,t) \leq \Delta_P(s,t) \leq (1+\varepsilon)d_P(s,t)$, where $d_P(s,t)$ is the length of the shortest path between s and t on ∂P. The algorithm also produces a polygonal path with $O(1/\varepsilon^{1.5})$ segments that avoids the interior of P and has length $\Delta_P(s,t)$.

Our second related result is: Given a convex polytope P with n edges in $\mathbb{R}^3$, and a parameter $0 < \varepsilon \leq 1$, we present an $O(n + 1/\varepsilon^5)$-time algorithm that computes two points $\mathfrak{s}, \mathfrak{t} \in \partial P$ such that $d_P(\mathfrak{s}, \mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_P$, where $\mathcal{D}_P = \max_{s,t \in \partial P} d_P(s,t)$ is the geodesic diameter of P.

1 Introduction

The *three-dimensional Euclidean shortest-path problem* is defined as follows: Given a set of pairwise-disjoint polyhedral objects in $\mathbb{R}^3$ and two points s and t, compute the shortest path between s and t which avoids the interiors of the given polyhedral 'obstacles'. This problem has received considerable attention in computational geometry. It was shown to be NP-hard by Canny and Reif [3], and the fastest available algorithms for this problem run in time that is exponential in the total number of obstacle vertices (which we denote by n) [15, 16]. The intractability of the problem has motivated researchers to develop polynomial-time algorithms for computing approximate shortest paths and for computing shortest paths in special cases.

In the *approximate three-dimensional Euclidean shortest-path* problem, we are given an additional parameter $\varepsilon > 0$, and the goal is to compute a path between s and t that avoids the interiors of the obstacles and whose length is at most $(1+\varepsilon)$ times the length of the shortest obstacle-avoiding path (we call such a path an ε-approximate path). Approximation algorithms for the three-dimensional shortest path problem were first studied by Papadimitriou [13], who

*A preliminary version of the paper to appear in the *13th ACM Symposium of Computational Geometry, 1997*. This work has been supported by a grant from the U.S.–Israeli Binational Science Foundation. This work is part of the author's Ph.D. thesis, prepared at Tel-Aviv University under the supervision of Prof. Micha Sharir.

†School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel; sariel@math.tau.ac.il

gave an $O(n^4(L + \log(n/\varepsilon))^2/\varepsilon^2)$-time algorithm for computing an ε-approximate shortest path, where L is the number of bits of precision in the model of computation. A rigorous analysis of Papadimitriou's algorithm was recently given by Choi et al. [5]. A different approach was taken by Clarkson [6], whose algorithm computes an ε-approximate shortest path in close to $O(n^2/\varepsilon^4 \log^{O(1)} n)$ time (the complexity of Clarkson's algorithm depends also on an additional accuracy parameter).

In a companion paper [9], we present an improved solution to an extended variant of the problem: Given a polyhedral environment V, and a point $s \in V$, and a parameter $\varepsilon > 0$, preprocess them into a data-structure that supports fast ε-approximate shortest path queries from s to any $t \in V$. The preprocessing takes roughly $O((n^4/\varepsilon^6)\log^2(n)\log^2(1/\varepsilon) + (n^2/\varepsilon^8)\log^5(1/\varepsilon))$ time and the size of the data structure is $O(n^2/\varepsilon^{4+\delta})$, for any $\delta > 0$ (the time complexity of the algorithm depends also on an additional accuracy parameter, since it uses Clarkson algorithm [6]), and a query can be answered in $O(\log(n/\varepsilon))$ time.

The problem of computing a shortest path between two points along the surface of a single convex polytope is an interesting special case of the three-dimensional Euclidean shortest-path problem. Sharir and Schorr [17] gave an $O(n^3 \log n)$ algorithm for this problem, exploiting the property that a shortest path on a polyhedron *unfolds* into a straight line segment. Mitchell et al. [12] improved the running time to $O(n^2 \log n)$; their algorithm works for non-convex polyhedra (or polyhedral surfaces) as well. Chen and Han [4] gave another algorithm with an improved running time of $O(n^2)$. It is a rather long-standing and intriguing open problem whether the shortest path on a convex polytope can be computed in subquadratic time. This has motivated the problem of finding near-linear algorithms that produce only an approximation of the shortest path. That is, we are given a convex polytope P with n vertices, two points s and t on its surface, and a positive real number ε. Let $\pi_P(s,t)$ denote any shortest path between s and t along the surface of P, and let $d_P(s,t)$ denote its length ($\pi_P(s,t)$ is usually, but not always, unique). We want to compute a path on the surface of P between s and t whose length is at most $(1+\varepsilon)d_P(s,t)$.

The first result in this direction is by Hershberger and Suri [11]. They present a simple algorithm that runs in $O(n)$ time, and computes a path whose length is at most $2d_P(s,t)$. Using the algorithm of [11], Agarwal et al. [2] present a relatively simple algorithm that computes an ε-approximate shortest path, this is a path on ∂P between the points $s, t \in \partial P$ whose length is at most $(1+\varepsilon)d_P(s,t)$, for any prescribed $0 < \varepsilon \le 1$, where the running time of the algorithm is $O(n\log(1/\varepsilon) + 1/\varepsilon^3)$.

In this paper we extend the analysis of [2] to solve two problems involving approximate shortest paths on convex polytopes in $\mathbb{R}^3$. In both problems, our solutions are much faster than the best known algorithms for the corresponding exact problems.

Approximate shortest paths queries. In Section 3.1, we present an alternative to the algorithm of [11] that receives as input a convex polytope P with n edges and preprocesses P in $O(n)$ time, such that for any $s, t \in \partial P$, one can compute, in $O(\log n)$ time, a number $\Delta_P(s,t)$, such that $d_P(s,t) \le \Delta_P(s,t) \le 8d_P(s,t)$. In Section 3.2, we modify the algorithm of [2], so that it uses the new algorithm instead of the algorithm of [11]. This results in an algorithm with $O(n)$ preprocessing time, such that for any $s, t \in \partial P$ and $0 < \varepsilon \le 1$, it computes a length $\Delta_P(s,t)$ of an ε-approximate shortest path between s and t on ∂P, in $O((\log n)/\varepsilon^{1.5} + 1/\varepsilon^3)$ time. In contrast, the fastest known algorithm for computing a data-structure that supports queries of computing the length of the exact shortest path between any pair of points on ∂P is due to [1]; it requires $O(n^6 m^{1+\delta})$ space and preprocessing time, for any $\delta > 0$, and answers a query in $O((\sqrt{n}/m^{1/4}) \log m)$ time, where $1 \leq m \leq n^2$ (the constants of proportionality depends on δ).

Approximate geodesic diameter. We present in Section 4 an algorithm that computes, for a given convex polytope P with n edges in $\mathbb{R}^3$, and a parameter $0 < \varepsilon \leq 1$, two points $\mathfrak{s}, \mathfrak{t} \in \partial P$ such that $d_P(\mathfrak{s}, \mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_P$, where $\mathcal{D}_P = \max_{s,t \in \partial P} d_P(s,t)$ is the *geodesic diameter* of P. The running time of the algorithm is $O(n + 1/\varepsilon^5)$. In contrast, the fastest known algorithm for the exact geodesic diameter takes $O(n^8 \log n)$ time (see [1]).

The paper is organized as follows. Section 2 introduces the required terminology and establishes some initial properties, mostly taken or adapted from [2]. In Section 3 we describe the algorithm for answering arbitrary two-point approximate shortest-path queries. In Section 4 we derive an algorithm for approximating the geodesic diameter of a convex polytope in $\mathbb{R}^3$. We conclude in Section 5 by mentioning a few open problems.

2 Preliminaries

We begin with some terminology and some initial observations, most of them taken or adapted from [2]. Let P be a convex polytope with n edges in $\mathbb{R}^3$. For a face f of P, we denote by H_f the plane passing through f. Given a set $A \subseteq \mathbb{R}^3$, and a real number $r \geq 0$, let A_r denote the *Minkowski sum* $A \oplus B_r$, where B_r is a ball of radius r about the origin; that is,

$$A_r = \left\{ x \;\middle|\; \inf_{y \in A} |x - y| \leq r \right\}.$$

For a convex polytope P and for $x \in \partial P$, let $F_{P,r}(x) = \left\{ y \;\middle|\; y \in \partial P_r, |x-y| = r \right\}$ be the set of points of ∂P_r 'corresponding' to x. See Figure 1 for an example of such an inflated polytope, and see [2] for (straightforward) details concerning the structure of P_r and of $F_{P,r}$.

For any plane H that avoids the interior of P, the *positive half-space* H^+ (resp. *negative half-space* H^-) bounded by H is the one containing P (resp. avoiding the interior of P); these half-spaces are assumed to be closed. Such a plane H is a *supporting plane* of P if $\partial P \cap H \neq \emptyset$.

Given a positive number $r \in \mathbb{R}$, we denote by $T_r(p) = r \cdot p$ the linear scaling of $\mathbb{R}^3$ by the factor r. Given two points $p, q \in \mathbb{R}^3$ such that $q \neq 0$, we denote by $ray(p,q)$ the ray emanating from p in the direction of q; that is, $ray(p,q) = \{p + tq | t \geq 0\}$.

Given two curves[1] γ_1, γ_2 in $\mathbb{R}^3$ that share an endpoint, we denote by $\gamma_1 \mid\mid \gamma_2$ the curve resulting from concatenating γ_1 to γ_2. For a curve $\gamma \subset \mathbb{R}^3$ and for $a, b \in \gamma$, we denote by $\gamma(a,b)$ the portion of γ between a and b. An *outer path* of a convex body K is a curve γ connecting two points on ∂K and disjoint from the interior of K. The length of a curve γ is denoted by $|\gamma|$.

The following well-known theorem implies that any outer path of P connecting two points $s, t \in \partial P$ must be of length at least $d_P(s,t)$, where, as above, $d_P(s,t)$ denotes the length of the shortest path between s and t on ∂P.

[1] By a curve we mean the image of a 1-1 continuous mapping of the unit interval $[0,1]$ into $\mathbb{R}^3$.

Theorem 2.1 ([14]) *Let F be the boundary of a convex body K, and γ a curve that does not meet the interior of K and connects points s and t on F. Then the length of γ is at least the shortest distance $d_K(s,t)$ between its endpoints along the surface F; it is strictly greater than this distance if the curve does not lie entirely on F.*

Lemma 2.2 ([2]) *Let P be a convex polytope in $\mathbb{R}^3$, let s,t be any two points on ∂P, let $0 < \varepsilon < 1$ be an arbitrary parameter, and let $r > 0$. For any $s_r \in F_{P,r}(s)$, $t_r \in F_{P,r}(t)$, there exists an outer path of P_r connecting s_r with t_r, whose length is at most $(1+\varepsilon/2)d_P(s,t)+2\pi r+100r/\sqrt{\varepsilon}$.*

Lemma 2.2 implies the following:

Lemma 2.3 *Let P,Q be convex polytopes in $\mathbb{R}^3$, let $0 < \varepsilon < 1$ be an arbitrary parameter, and let $r > 0$. If $P \subset Q \subset P_r$ then for any two points $s,t \in \partial P$ we have $d_P(s,t) \le d_Q(s',t') + 2r \le (1+\varepsilon/2)d_P(s,t) + (2\pi+4)r + 100r/\sqrt{\varepsilon}$, where $s' = ray(s,\eta_s) \cap \partial Q, t' = ray(t,\eta_t) \cap \partial Q$, and η_s, η_t denote the outward normals of P at s,t, respectively.*

Proof: Let $s_r = ray(s,\eta_s) \cap F_{P,r}(s)$, and $t_r = ray(t,\eta_t) \cap F_{P,r}(t)$. Clearly, $|ss_r| = |tt_r| = r$.

Let σ' be a shortest path connecting s_r to t_r on ∂P_r. By Lemma 2.2 and Theorem 2.1, we have $|\sigma'| \le (1+\varepsilon/2)d_P(s,t) + 2\pi r + 100r/\sqrt{\varepsilon}$. Let $\sigma = s's_r \mid\mid \sigma' \mid\mid t_rt'$. Clearly, $\sigma \cap int\, Q = \emptyset$, and $|\sigma| \le (1+\varepsilon/2)d_P(s,t) + (2\pi+2)r + 100r/\sqrt{\varepsilon}$.

By Theorem 2.1,

$$d_P(s,t) \le |ss'| + d_Q(s',t') + |t't| \le |\sigma| + 2r \le (1+\varepsilon/2)d_P(s,t) + (2\pi+4)r + 100r/\sqrt{\varepsilon}.$$

□

Let $P \subseteq \mathbb{R}^3$ be a convex polytope. A *point-normal pair* on P is an ordered pair (p,η_p) such that (1) $p \in \partial P$, (2) the plane $H(p,\eta_p)$ that passes through p and whose normal is η_p is a supporting plane of P, and (3) η_p is an outward normal to P at p. As above, we denote by $H^+(p,\eta_p)$ the closed halfspace containing P that is bounded by $H(p,\eta_p)$. For any $\delta > 0$, we call a collection S of point-normal pairs on P a δ-*dense* set if for any point-normal pair (q,η_q) on P, there exists a $(p,\eta_p) \in S$ such that $|pq| \le \delta$, and the angle between η_p and η_q is at most δ. Let $P(S)$ denote the intersection of all the halfspaces corresponding to elements of S, i.e., $P(S) = \bigcap_{(p,\eta_p)\in S} H^+(p,\eta_p)$.

Lemma 2.4 ([2]) *Given a convex polytope P in $\mathbb{R}^3$ having n edges and contained in a unit ball, then one can construct, in $O(n + (1/\varepsilon^2)\log(1/\varepsilon))$ time, an ε-dense set S of P, such that $|S| = O(1/\varepsilon^2)$, and $P \subseteq P(S) \subseteq P_{2\varepsilon^2}$.*

3 Approximate Shortest-Path Queries on a Convex Polytope

Let P be a given convex polytope in $\mathbb{R}^3$ with n edges. In this section we present an algorithm that preprocesses P in linear time, into a linear-size data structure, such that given any two query points $s,t \in \partial P$ and a parameter $0 < \varepsilon \le 1$, one can compute, in $O((\log n)/\varepsilon^{1.5} + 1/\varepsilon^3)$ time, an outer path connecting s to t whose length $\Delta_P(s,t)$ satisfies $d_P(s,t) \le \Delta_P(s,t) \le (1+\varepsilon)d_P(s,t)$. In Section 3.1, we describe an algorithm with linear preprocessing time that can answer, in $O(\log n)$ time, a factor 8 approximate shortest-path queries. In Section 3.2, we show how to plug this new algorithm into the algorithm of [2], resulting in the specified algorithm.

3.1 A Constant Factor Approximation Algorithm

Given two points $s, t \in \partial P$, we assume, without loss of generality, that $s = (0,0,-l)$ and $t = (0,0,l)$. For $r \ge 0$, let $B(r)$ denote the axis parallel cube of side $2r$ centered at the origin. Let $C(r) = \partial B(r) \setminus int\, P$. Let $s(r)$ and $t(r)$ denote the lower and upper intersection points of $C(r)$ with the z-axis, respectively. Note that these points are well-defined for $r \ge l$.

Lemma 3.1 *For any $r \ge l$, there exists a path σ between s and t on ∂P such that $\sigma \subset B(r)$, if and only if $s(r)$ and $t(r)$ are connected in $C(r)$.*

Proof: If the origin o lies on the boundary of P then $st \subseteq \partial P$ and the lemma follows easily. Otherwise, for any $x \in \partial B(r)$, let $f_r(x)$ denote the point $ray(o,x) \cap \partial P$. It easy to verify, that the function f_r is a continuous bijective deformation of $\partial B(r)$ onto ∂P. In particular, given a path $\sigma \subset B(r)$ connecting s to t on ∂P, then $\sigma^{-1} = f_r^{-1}(\sigma)$ is connected, $\sigma^{-1} \subset C(r)$, and $s' = f_r^{-1}(s), t' = f_r^{-1}(t) \in \sigma^{-1}$, implying that σ^{-1} connects s' to t' in $C(r)$.

Conversely, if $s(r)$ and $t(r)$ are connected in $C(r)$, then there exists a path γ on $C(r)$ connecting $s(r)$ to $t(r)$. Clearly, $f_r(\gamma)$ is connected, connects s to t on ∂P, and $f_r(\gamma) \subset B(r)$. $\square$

Consider the set $V = \{-1,0,1\} \times \{-1,0,1\} \times \{-1,1\}$ of 18 points on $B(1)$. Let $s' = (0,0,-1)$, $t' = (0,0,1)$. Define a set E of 28 straight segments, such that $vw \in E$ if and only if $v \ne w \in V$, and either $||v - w|| = 1$ or $x(v) = x(w) = \pm 1$, and $y(v) = y(w) = \pm 1$. The resulting graph $G = (V, E)$ is depicted in Figure 2. For $r \ge 0$, let $G(r) = (V, E(r))$, where $vw \in E(r)$ if $vw \in E$ and the segment $T_r(vw)$ does not intersect $int\, P$, where $T_r(a) = ra$ is the linear scaling of $\mathbb{R}^3$ by the factor r.

Clearly, $G(r)$ is monotone increasing, that is, $G(r') \subseteq G(r)$, for any $0 \le r' \le r$. Moreover, $G(0) = (V, \emptyset)$ and $G(r) = G$, for r sufficiently large.

Lemma 3.2 *For $r \ge l$, the vertices s', t' in $G(r)$ are connected if and only if $s(r)$ and $t(r)$ are connected in $C(r)$.*

Proof: If s' and t' are connected by a path σ in $G(r)$, then the path $T_r(\sigma)$ connects $s(r)$ to $t(r)$, and satisfies $T_r(\sigma) \subset C(r)$.

Conversely, suppose that $s(r)$ and $t(r)$ are connected in $C(r)$. Let $G_r = T_r(G)$ denote the natural embedding of the graph G into the boundary of $B(r)$.

Let σ be a path connecting $s(r)$ to $t(r)$ in $C(r)$ (since $C(r)$ is a polyhedral surface, we may assume that σ intersect each edge of G_r a finite number of times). For each face f of G_r (f is an axis-parallel square of side 1 or 2), such that $\sigma \cap int\, f \ne \emptyset$, let p_f and q_f be two intersection points of σ with ∂f, such that $int\, \sigma(p_f, q_f) \subseteq int\, f$. Partition ∂f into two Jordan arcs γ_1, γ_2 with disjoint relative interiors such that the points p_f and q_f are their common endpoints.

By the convexity of P, it follows that either γ_1 or γ_2 avoids the interior of P, because $\sigma(p_f, q_f) \cap int\, P = \emptyset$. Assume, without loss of generality, that γ_1 avoids the interior of P

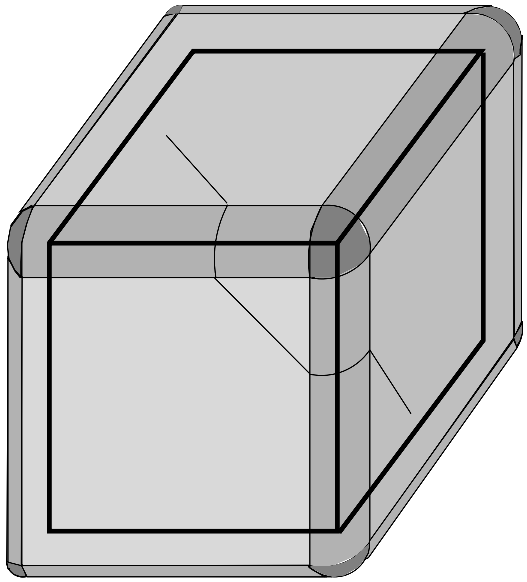

Figure 1: A cube P and the inflated set P_r

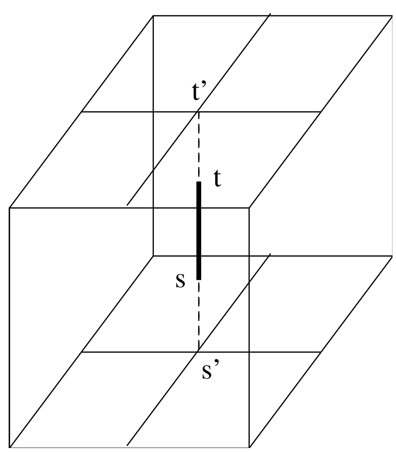

Figure 2: The graph G defined on the boundary of the cube $B(1)$

(see Figure 3). We then replace σ by the curve $\sigma(s(r), p_f) \mid\mid \gamma_1 \mid\mid \sigma(q_f, t(r))$. We repeat this replacement process for each connected portion of $\sigma \cap int\, f$, until $\sigma \cap int\, f = \emptyset$, and apply this procedure for all faces f of G_r. Let σ' be the resulting path.

Clearly, σ' lies completely on the edges of G_r, and it connects $s(r)$ to $t(r)$. Moreover, σ' avoids the interior of P. This implies that $s(r)$ and $t(r)$ are connected in G_r, by a path that avoids the interior of P. Thus s' and t' are connected in $G(r)$. □

It is easy to verify the following:

Lemma 3.3 *The length of the shortest path between s' and t' in any subgraph of G in which s' and t' are connected is at most 18, where all the horizontal edges of G have weight 1 and all the vertical edges have weight 2.*

Actually, the subgraphs $G(\cdot)$ of G have additional conditions imposed on them because of the convexity of P. By checking all possible subgraphs of G that comply with these "convexity" conditions, we get the following improvement, which only effects the constants of proportionality in the resulting algorithm.

Lemma 3.4 *The length of the shortest path between s' and t' in any subgraph $G(r)$ of G in which s' and t' are connected is at most 14, where the edges of G have the same weights as above.*

Moreover, there exist a convex polytope P, a pair of points $s, t \in \partial P$ and $r > 0$ such that the length of the shortest path between s' and t' in $G(r)$ is 14.

The upper bound follows from an exhaustive search through all possible subgraphs $G(r)$ that we performed by a computer program. This can be also verified manually, but it is rather tedious to do so. The lower bound is depicted in Figure 4. The polytope P is the convex hull of all the black dots in that figure (in fact, we have to inflate P a little so that the black dots lie in its interior).

Lemma 3.5 *Let r^* be the minimal value of r for which s' and t' belong to the same connected component of $G(r)$. Then $2r^* \le d_P(s,t) \le 16r^*$.*

Proof: Let γ' be the path on $C(r^*)$ corresponding to the shortest path between s' and t' in $G(r^*)$. By Lemma 3.4, the length of γ' is at most $14r^*$. Let $\gamma = ss(r^*) \mid\mid \gamma' \mid\mid t(r^*)t$. Clearly, $|\gamma| \le 16r^*$. Since γ is an outer path of P connecting s to t, Theorem 2.1 implies that $d_P(s,t) \le 16r^*$.

Lemma 3.2 and Lemma 3.1 imply that there exists a path $\sigma' \subset B(r)$ connecting s to t on ∂P, if and only if s' and t' are connected in $G(r)$.

Let σ be a shortest path between s and t on ∂P. We claim that σ must intersect $\partial B(r^*)$, for otherwise s' and t' are connected in $G(r^*)$, and this also holds for $r' < r^*$ sufficiently close to it. Hence the vertices s' and t' are connected in $G(r')$, a contradiction to the choice of r^*.

If σ intersects one of the vertical faces of $B(r^*)$ then its length is at least $2r^*$. If σ intersects the top or bottom face of $B(r^*)$, then its length is also at least $(r^* - l) + (r^* + l) = 2r^*$. □

Lemma 3.6 *For an edge $e \in E$, let $r(e)$ denote the minimal value of r for which $e \in E(r)$. Let $e_1, \ldots, e_{28}$ be the edges of E ordered in increasing value of $r(\cdot)$. Then $r^* = r(e_i)$, where i is the smallest index for which s' and t' are connected in the graph $G(r(e_i)) = (V, \{e_1, \ldots, e_i\})$.*

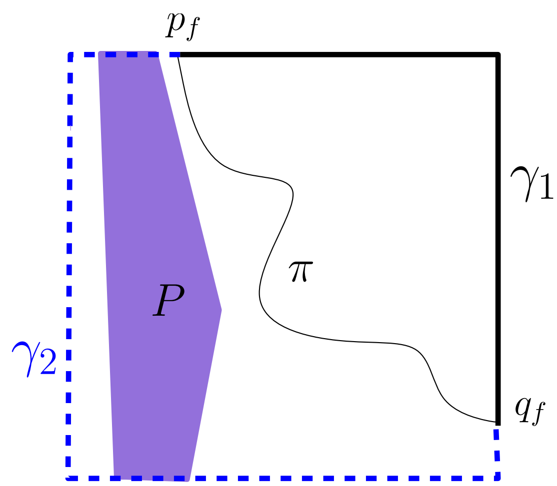

Figure 3: Either γ_1 or γ_2 must avoid the interior of P.

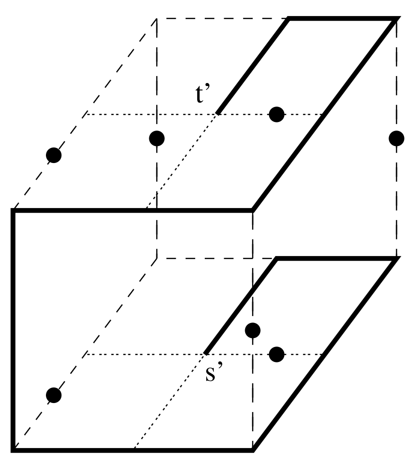

Figure 4: A path realizing the longest possible shortest path (of length 14) in a subgraph $G(r)$.

Proof: Obvious. □

Lemma 3.7 *A convex polytope P with n edges in $\mathbb{R}^3$ can be preprocessed in linear time, into a linear-size data structure, such that given any two points $s, t \in \partial P$, one can calculate, in $O(\log n)$ time, a value $\Delta_P(s,t)$ satisfying $d_P(s,t) \le \Delta_P(s,t) \le 8d_P(s,t)$.*

Proof: We construct, in $O(n)$ time, the Dobkin-Kirkpatrick hierarchical decomposition of P [7, 8].

By Lemma 3.6, all we have to do is to compute, for each edge $e = pq \in E$, the value of $r(e)$, which is the minimal value of r for which $e \in G(r)$. Once these values are available, r^* can be obtained in $O(1)$ time. (Recall that the graphs $G(r)$ depend on s and t.)

The value of $r(e)$ is the largest r such that $T_r(e) \cap P \neq \emptyset$. Thus, using the hierarchical decomposition of P, it is now possible to compute, in $O(\log n)$ time, the two intersection points x_1, x_2 between ∂P and the rays induced by $T_r(p)$ and $T_r(q)$ as r varies. Moreover, by locally inspecting $\partial P \cap H$ near x_1, x_2, it can be decided whether any of these points is the last intersection point of $T_r(e)$ with ∂P, where H is the plane spanned by the origin and e. If not, the required intersection point lies between x_1 and x_2 on $\partial(P \cap H)$. We can find this intersection point by performing an extreme-point query on $P \cap H$. Namely, we are looking for the maximal r for which $l(r) \cap P \neq \emptyset$, where $l(r)$ is the line passing through $T_r(e)$. See Figure 5.

This can be easily done in $O(\log n)$ time if we are given the hierarchical decomposition of the planar polygon $P \cap H$. This two-dimensional hierarchical decomposition (of $P \cap H$) is stored implicitly in the three-dimensional hierarchical decomposition of P, and the query can be performed in $O(\log n)$ time. See [8] for details.

Thus, we can compute $r(e)$ in $O(\log n)$ time, for each edge e of G. Hence, as already noted, we can calculate r^* in $O(\log n)$ time.

We return $16r^*$ as our approximation value $\Delta_P(s,t)$. By Lemma 3.5, $2r^* \le d_P(s,t) \le 16r^*$, from which lemma follows. □

Remark 3.8 A constant factor approximation algorithm, similar to the algorithm of Lemma 3.7, was developed independently by Hershberger and Suri [10].

3.2 An $(1+\varepsilon)$-Approximation Algorithm

Given a convex polytope P with n edges, two points $s, t \in \partial P$, and a parameter $0 < \varepsilon \le 1$, the algorithm of [2] computes an outer path of P connecting s and t and approximating the shortest path, as follows (see [2] for more details):

- Compute a crude approximation d satisfying $d_P(s,t) \le d \le 2d_P(s,t)$, using the algorithm of [11]. This stage takes $O(n)$ time.
- Compute, in $O(n)$ time, the Dobkin-Kirkpatrick hierarchical decomposition of P [7, 8].
- Using d and the hierarchical decomposition of P, compute, in $O((\log n + \log(1/\varepsilon))/\varepsilon^{1.5})$ time, an approximation polytope $Q \supseteq P$ having $O(1/\varepsilon^{1.5})$ faces, such that $s, t \in \partial Q$ and $d_P(s,t) \le d_Q(s,t) \le (1+\varepsilon)d_P(s,t)$.

- Using the algorithm of [4], compute, in $O(1/\varepsilon^3)$ time, the shortest path on ∂Q between s and t. This is the desired approximating outer path.

A more detailed inspection of the algorithm of [2] reveals that any approximation d, satisfying $d_P(s,t) \leq d \leq c \cdot d_P(s,t)$ for some prescribed constant $c > 1$, can be used in the first stage of the algorithm. Thus, by moving the computation of the Dobkin-Kirkpatrick decomposition of P into the preprocessing stage, and by using the algorithm of Lemma 3.7 instead of the algorithm of [11], we get:

Theorem 3.9 *A given convex polytope P with n edges can be preprocessed in $O(n)$ time, such that given any two points s and t on ∂P, and a parameter $\varepsilon > 0$, one can construct a polygonal outer path of P between s and t, whose length is at most $(1+\varepsilon)d_P(s,t)$, and which consists of $O(1/\varepsilon^{1.5})$ segments, in time $O((\log n)/\varepsilon^{1.5} + 1/\varepsilon^3)$.*

Remark 3.10 The outer path σ of P generated by the algorithm of Theorem 3.9 can be projected onto ∂P in $O(n \log(1/\varepsilon) + 1/\varepsilon^3)$ time, resulting in a path σ' on ∂P, such that $|\sigma'| \leq |\sigma|$. See [2]. Since there are (easy) cases where any path on ∂P between s and t has $\Omega(n)$ edges, there is no hope of constructing approximate shortest path on ∂P itself in worst-case sublinear query time.

Remark 3.11 Given a convex polytope P with n edges, the fastest known algorithm for computing a data-structure that supports queries of computing the length of the exact shortest path between any pair of points on ∂P is due to [1]; it requires $O(n^6 m^{1+\delta})$ space and preprocessing time, and answers a query in $O((\sqrt{n}/m^{1/4})\log m)$ time, where $1 \leq m \leq n^2$, $\delta > 0$ (and the constants of proportionality depends on δ).

4 Approximating the Geodesic Diameter of a Convex Polytope

In this section we present an efficient algorithm that approximates the geodesic diameter of a convex polytope in $\mathbb{R}^3$.

Definition 4.1 *(i) Given a convex polytope P in $\mathbb{R}^3$, we define the* geodesic diameter *of P to be*

$$\mathcal{D}_P = \max_{s,t \in \partial P} d_P(s,t).$$

(ii) Given a subset $S \subseteq \partial P$, we denote by $\Delta_P(S)$ the geodesic diameter of S on the boundary of P, namely

$$\Delta_P(S) = \max_{s,t \in S} d_P(s,t).$$

(iii) Let $C(P)$ denote the smallest axis-parallel cube bounding P, and define the outer diameter *of P to be $\delta(P) = 6d$, where d is the side length of $C(P)$.*

Lemma 4.2 *Given a convex polytope P with n edges in $\mathbb{R}^3$, the outer diameter $\delta(P)$ of P can be calculated in linear time. Moreover, $\delta(P)/6 \leq \mathcal{D}_P \leq \delta(P)$.*

Proof: Clearly, $C(P)$, and thus also $\delta(P)$, can be computed in linear time. Let $d = \delta(P)/6$ be the edge length of $C(P)$. Since ∂P must contain two points s, t on opposite facets of $C(P)$, the Euclidean distance between s and t is $\geq d$, implying that $\mathcal{D}_P \geq d_P(s,t) \geq d$. It is also easy to verify that any pair of points on the boundary of P can be connected by an outer path of P of length at most $\delta(P) = 6d$. It follows that $\delta(P)/6 = d \leq \mathcal{D}_P \leq \delta(P)$. $\square$

The approximation algorithm is presented in Figure 6. The following Lemma prove the correctness of the algorithm, and describe it in more detail.

Lemma 4.3 *Given a convex polytope $\mathcal{P}$ with n edges in $\mathbb{R}^3$ such that $1/6 \leq \mathcal{D}_\mathcal{P} \leq 1$, and a parameter $0 < \varepsilon \leq 1$, then a pair of points $\mathfrak{s}, \mathfrak{t} \in \partial\mathcal{P}$ can be calculated, in $O(n + 1/\varepsilon^5)$ time, such that $d_\mathcal{P}(\mathfrak{s}, \mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_\mathcal{P}$.*

Proof: Let S be an $(\varepsilon/100)$-dense set of $\mathcal{P}$, and let S' be an $(\varepsilon^{3/4}/100)$-dense set of $\mathcal{P}$. By Lemma 2.4, such sets can be constructed in $O(n + (1/\varepsilon^2)\log(1/\varepsilon))$ time. Let s, t be two points on $\partial\mathcal{P}$ realizing the geodesic diameter of $\mathcal{P}$. Thus, there are two points $(s', \eta_{s'}), (t', \eta_{t'}) \in S$ such that $|ss'|, |tt'| \leq \varepsilon/100$ and $e(\eta_s, \eta_{s'}), e(\eta_t, \eta_{t'}) \leq \varepsilon/100$, where η_s, η_t are vectors normal to $\mathcal{P}$ at s, t, respectively, and $e(a, b)$ denotes the angle between the vectors a and b.

We claim that $d_\mathcal{P}(s, s'), d_\mathcal{P}(t, t') \leq \varepsilon/50$. Indeed, put $x = |ss'|$ and $\psi = e(\eta_s, \eta_{s'})$. Draw the planes H, H' that support $\mathcal{P}$ at s, s', and have normals η_s, $\eta_{s'}$, respectively. Let W be the wedge bounded by H and H' and containing $\mathcal{P}$. Let σ denote the shortest path on ∂W between s and s'. This path consists of two segments su, us', with a common endpoint u lying on $H \cap H'$. Moreover, the angle $\alpha = \measuredangle sus'$ is at least the obtuse dihedral angle between H and H'. That is $\alpha \geq \pi - \psi$. See Figure 7.

Put $y = |su|$, $z = |us'|$. By the cosine law,

$$x^2 = y^2 + z^2 - 2yz\cos\alpha = y^2 + z^2 + 2yz\cos(\pi - \alpha) \geq y^2 + z^2 + 2yz\cos\psi \geq (y+z)^2\cos\psi.$$

Since $\psi \leq \varepsilon/100 \leq 1/100$, we have $\cos\psi \geq 1/4$, so $y + z \leq 2x$. Since $\sigma = sus'$ is an outer path of $\mathcal{P}$ between s and s', we have $d_\mathcal{P}(s, s') \leq 2|ss'| \leq \varepsilon/50$.

Thus,

$$\mathcal{D}_\mathcal{P} = d_\mathcal{P}(s,t) \leq d_\mathcal{P}(s,s') + d_\mathcal{P}(s',t') + d_\mathcal{P}(t',t) \leq \varepsilon/25 + \Delta_\mathcal{P}(S).$$

It follows that $\mathcal{D}_\mathcal{P} \leq \varepsilon/25 + \Delta_\mathcal{P}(S) \leq (\varepsilon/2)\mathcal{D}_\mathcal{P} + \Delta_\mathcal{P}(S)$, or

$$\Delta_\mathcal{P}(S) \geq (1 - \varepsilon/2)\mathcal{D}_\mathcal{P}. \tag{1}$$

Let $Q = P(S') = \bigcap_{(p,\eta_p)\in S'} H^+(p, \eta_p)$. The complexity of Q is $O(1/\varepsilon^{3/2})$, and, by Lemma 2.4, $Q \subset \mathcal{P}_r$, for $r = \varepsilon^{3/2}/5000$.

For $s \in S$, let $\mu_Q(s)$ denote the point $ray(s, \eta_s) \cap \partial Q$, where η_s denotes the outward normal to $\mathcal{P}$ at s. The mapping $\mu_Q(\cdot)$ can be computed in $O((1/\varepsilon^2)\log(1/\varepsilon))$ time, by preprocessing Q for ray shooting by computing, in $O(1/\varepsilon^{3/2})$ time, the Dobkin-Kirkpatrick hierarchical decomposition of Q [7, 8]. For $p \in S$, one can use this decomposition to compute, in $O(\log(1/\varepsilon))$ time, the intersection point between $ray(p, \eta_p)$ and ∂Q.

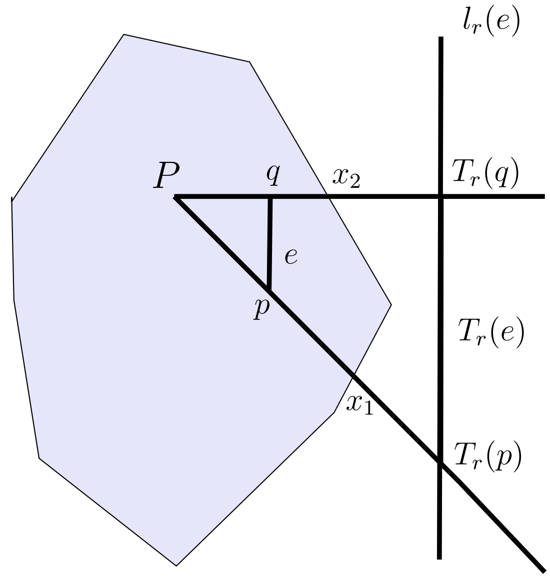

Figure 5: Computing the minimal value for which edge e appears in $G(r)$.

ALGORITHM APPROXIMATE-GEODESIC-DIAMETER(P, ε)

`Input:` A convex polytope P, approximation factor ε

`Output:` Two points $\mathfrak{s}, \mathfrak{t}$ on the boundary of P, such that $d_P(\mathfrak{s}, \mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_P$

begin

Compute the outer diameter of P: $\delta = \delta(P)$

$\mathcal{P} \leftarrow T_{1/\delta}(P)$, where $T_{1/\delta}(\cdot)$ us the linear scaling of $\mathbb{R}^3$ by the factor $1/\delta$

$S \leftarrow DenseSet(\mathcal{P}, \varepsilon/100)$, $S' \leftarrow DenseSet(\mathcal{P}, \varepsilon^{3/4}/100)$

$Q = P(S') = \bigcap_{(p,\eta_p)\in S'} H^+(p, \eta_p)$

Project the points of S to the boundary of Q:

$$\mu_Q(S) \leftarrow \left\{ \mu_Q(s) \;\middle|\; s \in S \right\}, \text{ where } \mu_Q(s) = ray(s, \eta_s) \cap \partial Q$$

for each $s \in \mu_Q(S)$ do

Compute the distance $d_Q(s,t)$, for all $t \in \mu_Q(S)$)

end for

Let $\mathfrak{s}, \mathfrak{t}$ be the pair of points of S for which $d_Q(\mu_Q(\mathfrak{s}), \mu_Q(\mathfrak{t}))$ is maximal

return $T_\delta(\mathfrak{s})$, $T_\delta(\mathfrak{t})$

end APPROXIMATE-GEODESIC-DIAMETER

Figure 6: Algorithm for computing an approximated geodesic diameter of a polytope.

Applying Lemma 2.3, with this value of r and with $\varepsilon/2$, we obtain

$$\begin{aligned} d_{\mathcal{P}}(s,t) &\leq d_Q(\mu_Q(s),\mu_Q(t)) + 2r \leq (1+\varepsilon/4)d_{\mathcal{P}}(s,t) + (1/400)\varepsilon^{3/2} + (\sqrt{2}/50)\varepsilon \\ &\leq d_{\mathcal{P}}(s,t) + \left(\frac{1}{4}\varepsilon + \frac{6}{400}\varepsilon^{3/2} + \frac{6\sqrt{2}}{50}\varepsilon\right)\mathcal{D}_{\mathcal{P}} \leq d_{\mathcal{P}}(s,t) + (\varepsilon/2)\mathcal{D}_{\mathcal{P}}, \end{aligned}$$

for any $s, t \in S$.

Let $\mathfrak{s}, \mathfrak{t}$ be two points in S for which $d = d_Q(\mu_Q(\mathfrak{s}), \mu_Q(\mathfrak{t}))$ is maximal. The above inequality implies

$$\Delta_{\mathcal{P}}(S) \leq d_Q(\mu_Q(\mathfrak{s}),\mu_Q(\mathfrak{t})) + 2r \leq d_{\mathcal{P}}(\mathfrak{s},\mathfrak{t}) + (\varepsilon/2)\mathcal{D}_{\mathcal{P}}.$$

Hence, by (1),

$$d_{\mathcal{P}}(\mathfrak{s},\mathfrak{t}) \geq \Delta_{\mathcal{P}}(S) - (\varepsilon/2)\mathcal{D}_{\mathcal{P}} \geq (1-\varepsilon/2)\mathcal{D}_{\mathcal{P}} - (\varepsilon/2)\mathcal{D}_{\mathcal{P}} = (1-\varepsilon)\mathcal{D}_{\mathcal{P}}.$$

As for the time complexity, the distances from a fixed point $\mu_Q(p)$ to all other points of $\mu_Q(S)$ along Q, can be calculated, in $O(1/\varepsilon^3)$ time, by the algorithm of [4]. Thus, the pair of points realizing d can be calculated in $O(1/\varepsilon^5)$ time. □

Theorem 4.4 *Given a convex polytope P in $\mathbb{R}^3$ with n edges, and a parameter $0 < \varepsilon \leq 1$, one can compute, in $O(n + 1/\varepsilon^5)$ time, a pair of points $\mathfrak{s}, \mathfrak{t} \in \partial P$ and a value Δ, such that $d_P(\mathfrak{s},\mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_P$ and $(1-\varepsilon)\mathcal{D}_P \leq \Delta \leq (1+\varepsilon)\mathcal{D}_P$.*

Proof: Compute in linear time the outer diameter $\delta = \delta(P)$. Let $\mathcal{P} = T_{1/\delta}(P)$, where $T_{1/\delta}(\cdot)$ is the linear scaling of $\mathbb{R}^3$ by the factor $1/\delta$. By Lemma 4.2, $1/6 \leq \mathcal{D}_{\mathcal{P}} \leq 1$.

By Lemma 4.3, a pair of points $\mathfrak{s}', \mathfrak{t}'$ of $\partial\mathcal{P}$ can be calculated in $O(n + 1/\varepsilon^5)$ time, such that

$$d_{\mathcal{P}}(\mathfrak{s}',\mathfrak{t}') \geq (1-\varepsilon)\mathcal{D}_{\mathcal{P}}. \tag{2}$$

Let $\mathfrak{s}, \mathfrak{t}$ be the points on ∂P corresponding to $\mathfrak{s}', \mathfrak{t}'$, respectively. Then, by multiplying (2) by δ, it follows that $d_P(\mathfrak{s},\mathfrak{t}) \geq (1-\varepsilon)\mathcal{D}_P$.

Using the algorithm of [2], we can compute in $O(n + 1/\varepsilon^3)$ time an ε-approximation Δ to $d_P(\mathfrak{s},\mathfrak{t})$, such that $(1-\varepsilon)\mathcal{D}_P \leq \Delta \leq (1+\varepsilon)\mathcal{D}_P$. □

Remark 4.5 The fastest exact algorithm for computing the geodesic diameter of a convex polytope in $\mathbb{R}^3$ is due to Agarwal et al. [1], and runs in $O(n^8 \log n)$ time.

5 Conclusions

In this paper we have presented two simple and efficient algorithms for computing approximate solutions to problems involving shortest paths on the surface of a convex polytope in 3-space. We conclude by mentioning a few open problems.

- Can an ε-approximate shortest path between two points on a polyhedral terrain, or on the surface of a nonconvex polyhedron, be computed in time that is near-linear in the number of edges? A recent subquadratic solution has been announced by Varadarajan and Agarwal [18].

- Can the exact shortest path between two points on a convex polyhedron be computed in near-linear time? in subquadratic time?
- A geodesic path on a convex polytope P is a path which is locally optimal. (i.e., it can not be made shorter by a "local" shortcut; in other words, it is a polygonal path with vertices only on edges of P, so that the angles between two consecutive segments and the edge of P to which they are adjacent are equal.) Clearly, a shortest path is geodesic. Can one compute a geodesic path between two points on a convex polyhedron in subquadratic time?

Acknowledgments

The author wishes to thank Pankaj Agarwal and Micha Sharir for helpful discussions concerning the problems studied in this paper and related problems.

The author also wish to thank the referees for their comments and suggestions.

References

[1] P. K. Agarwal, B. Aronov, J. O'Rourke, and C. A. Schevon. Star unfolding of a polytope with applications. *SIAM J. Comput.*, 26:1679–1713, 1997.

[2] P. K. Agarwal, S. Har-Peled, M. Sharir, and K. R. Varadarajan. Approximate shortest paths on a convex polytope in three dimensions. *J. Assoc. Comput. Mach.*, 44:567–584, 1997.

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

[4] J. Chen and Y. Han. Shortest paths on a polyhedron; Part I: computing shortest paths. *Int. J. Comput. Geom. & Appl.*, 6(2):127–144, 1996.

[5] J. Choi, J. Sellen, and C. K. Yap. Approximate Euclidean shortest path in 3-space. In *Proc. 10th Annu. ACM Sympos. Comput. Geom.*, pages 41–48, 1994.

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

[7] D. P. Dobkin and D. G. Kirkpatrick. A linear algorithm for determining the separation of convex polyhedra. *J. Algorithms*, 6:381–392, 1985.

[8] D. P. Dobkin and D. G. Kirkpatrick. Determining the separation of preprocessed polyhedra – a unified approach. In *Proc. 17th Internat. Colloq. Automata Lang. Program.*, volume 443 of *Lect. Notes in Comp. Sci.*, pages 400–413. Springer-Verlag, 1990.

[9] S. Har-Peled. Constructing approximate shortest path maps in three dimensions. *SIAM J. Comput.*, 28(4):1182–1197, 1999.

[10] J. Hershberger. personal communication. 1997.

[11] J. Hershberger and S. Suri. Practical methods for approximating shortest paths on a convex polytope in $\mathbb{R}^3$. In *Proc. 6th ACM-SIAM Sympos. Discrete Algorithms*, pages 447–456, 1995.

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

[13] C. H. Papadimitriou. An algorithm for shortest-path motion in three dimensions. *Inform. Process. Lett.*, 20:259–263, 1985.

[14] A. V. Pogorelov. *Extrinsic Geometry of Convex Surfaces*, volume 35 of *Translations of Mathematical Monographs*. Amer. Math. Soc., 1973.

[15] J. H. Reif and J. A. Storer. A single-exponential upper bound for finding shortest paths in three dimensions. *J. Assoc. Comput. Mach.*, 41(5):1013–1019, 1994.

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

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

[18] K. R. Varadarajan and P. K. Agarwal. Approximating shortest paths on a nonconvex polyhedron. In *Proc. 38th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 182–191, 1997.

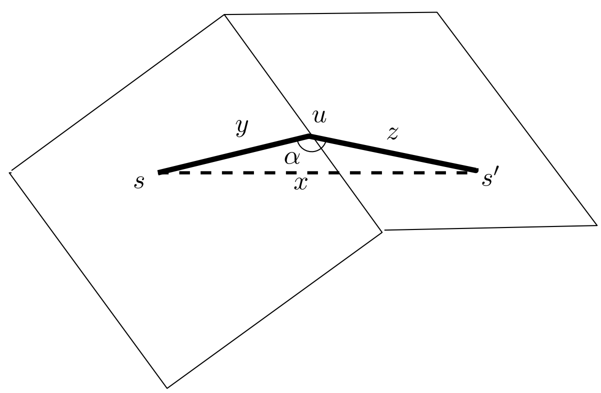

Figure 7: $y + z \leq 2x$.