

Flying over a Polyhedral Terrain

Hamid Zarrabi-Zadeh

School of Computer Science, University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
hzarrabi@uwaterloo.ca

Abstract. We consider the problem of computing shortest paths in three-dimensions in the presence of a single-obstacle polyhedral terrain, and present a new algorithm that for any $p \geq 1$, computes a $(c + \varepsilon)$ -approximation to the L_p -shortest path above a polyhedral terrain in $O(\frac{n}{\varepsilon} \log n \log \log n)$ time and $O(n \log n)$ space, where n is the number of vertices of the terrain, and $c = 2^{(p-1)/p}$. This leads to a FPTAS for the problem in L_1 metric, a $(\sqrt{2} + \varepsilon)$ -factor approximation algorithm in Euclidean space, and a 2-approximation algorithm in the general L_p metric.

1 Introduction

Computing shortest paths in geometric domains is a fundamental problem in computational geometry, having various applications in robot motion planning and navigation systems. There is a large body of work in this area, a broad overview of which can be found in the surveys by Mitchell [6, 7].

The problem of computing a two-dimensional shortest path among a set of polygonal obstacles is widely studied, and there are algorithms [4] solving the problem in the Euclidean metric (or in any L_p metric, $p \geq 1$) in optimal running time $O(n \log n)$, where n is the total number of vertices of the polygonal obstacles. In three-dimensions, the problem of computing a shortest path among a set of polyhedral obstacles is well-known to be NP-hard [2] even in L_1 metric. However, for several classes of obstacles, the L_1 -shortest path can be computed in polynomial time. For instance, if the obstacle is a single “polyhedral terrain” then the L_1 -shortest path can be computed exactly in $O(n^3 \log n)$ time [8].

The first ε -approximation algorithm for the 3D Euclidean shortest path problem is given by Papadimitriou [9] with running time $O(n^4 \varepsilon^{-2} (N + \log \frac{n}{\varepsilon})^2)$, where n is the total number of vertices of the polyhedral obstacles, and N is the maximum bit-length of the input integers. A different approach was taken by Clarkson [3] resulting in an algorithm which is faster when $n\varepsilon^3$ is large. Asano *et al.* [1] have slightly improved the running time of Papadimitriou’s algorithm to $O(n^4 \varepsilon^{-2} \log N)$.

In this paper, we consider the 3D shortest path problem in the presence of a single-obstacle polyhedral terrain, while distances are computed in general L_p metric ($p \geq 1$). The problem definition is as follows:

Problem Given an n -vertex polyhedral terrain T and two points s and t on or above T , find the L_p -shortest path from s to t that fully stays on or above T .

Each input coordinate is assumed to be represented using a rational number whose numerator and denominator are integers of bit-length at most N .

We present an efficient algorithm that computes a $(1 + \varepsilon)$ -approximation to the L_1 -shortest path above a polyhedral terrain in $O(\frac{n}{\varepsilon} \log n \log \log n)$ time and $O(n)$ space. As mentioned earlier, there is an exact algorithm for the problem in L_1 metric requiring $O(n^3 \log n)$ time [8]. However, in practical applications the input terrain is an approximation of the reality. Therefore, exact solutions are often meaningless, and efficient approximation algorithms are usually preferred.

In general L_p metric, our algorithm computes a factor- $(2^{(p-1)/p} + \varepsilon)$ approximation to the L_p -shortest path above a polyhedral terrain in $O(\frac{n}{\varepsilon} \log n \log \log n)$ time and $O(n \log n)$ space. This gives a $(\sqrt{2} + \varepsilon)$ -approximation algorithm for the problem in the Euclidean space. Furthermore, by picking ε appropriately, we will guarantee that the length of the approximate path is at most twice the length of the optimal path in any L_p metric, $p \geq 1$.

2 Preliminaries and Properties

Let T be a polyhedral terrain with n vertices, and let s and t be two points on or above T . We assume without loss of generality that $z(s) = 0$ and $z(t) \geq 0$. Let π_{opt} be the L_p -shortest path between s and t that fully stays on or above T . We note that π_{opt} is not necessarily unique. For a path π in the L_p metric, the length of π denoted by $\|\pi\|_p$. Throughout this paper, we assume that p is a fixed parameter. Therefore, we suppress p in our notations and simply write $\|\pi\|$ instead of $\|\pi\|_p$.

Consider the plane $w(h) : z = h$. The intersection of T and $w(h)$ partitions $w(h)$ into a free region, $\mathcal{F}(h)$, and an obstacle region, $w(h) \setminus \mathcal{F}(h)$, as shown in Fig. 1. ($\mathcal{F}(h)$ consists of those points on $w(h)$ that lie on or above T). We denote by $s(h)$ and $t(h)$ the vertical projection of s and t on $w(h)$, respectively. Let $\bar{\pi}(h)$ be the L_p -shortest path from $s(h)$ to $t(h)$ that lies completely in $\mathcal{F}(h)$.

For $h \geq z(t)$, we construct a path from s to t above T as follows: we first move from s along a vertical segment to $s(h)$, then proceed from $s(h)$ to $t(h)$ along the planar path $\bar{\pi}(h)$, and finally descend from $t(h)$ along a vertical segment to t . We call such a vertical-horizontal-vertical path a *VHV-path with height h* and

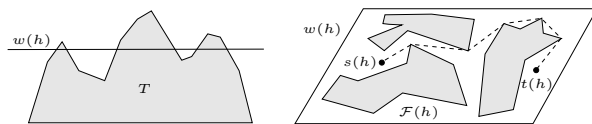


Fig. 1. The intersection of terrain T and plane $w(h)$.

denote it by $\pi(h)$. Among all VHV-paths above T , we refer to the one with the minimum L_p -length as the *optimal VHV-path above T* and denote it by π^* . We note again that π^* is not necessarily unique.

Mitchell and Sharir [8] have observed that in L_1 metric, the optimal VHV-path and the L_1 -shortest path above a polyhedral terrain have the same L_1 -length. We can generalize this fact to any L_p metric as follows:

Observation 1 *Let π^* be the optimal VHV-path, and π_{opt} be the L_p -shortest path above a polyhedral terrain. Then $\|\pi^*\| \leq 2^{(1-\frac{1}{p})} \|\pi_{\text{opt}}\|$.*

Proof. Suppose that π_{opt} is composed of k segments $s_i = (a_i, b_i)$. We use x_i, y_i and z_i to refer to the length of the projection of s_i in the x -, y - and z -direction, respectively. Let c_i be the vertical projection of a_i on a horizontal plane passing through b_i , and define $\sigma_i = (b_i, c_i)$ and $h_i = (a_i, c_i)$. Then

$$\frac{\|\sigma_i\| + \|h_i\|}{\|s_i\|} = \frac{(x_i^p + y_i^p)^{\frac{1}{p}} + z_i}{(x_i^p + y_i^p + z_i^p)^{\frac{1}{p}}} = \frac{\omega_i + z_i}{(\omega_i^p + z_i^p)^{\frac{1}{p}}}$$

where $\omega_i = (x_i^p + y_i^p)^{1/p}$. The above quotient is maximized when $\omega_i = z_i$, and hence, it is at most $\frac{2\omega_i}{\sqrt[p]{2\omega_i^p}} = 2^{(1-\frac{1}{p})}$.

Let h be the maximum z -coordinate of a point on π_{opt} . It is clear that $\|\pi^*\| \leq \|\pi(h)\|$. The path π_{opt} can be decomposed into two (possibly empty) z -monotone subpaths π^+ and π^- , where π^+ is ascending and π^- is descending in z -direction. Therefore, $\sum_i \|h_i\| = 2h - z(t)$. Furthermore, the vertical projections of σ_i 's on $w(h)$ form a path from $s(h)$ to $t(h)$ that completely lies in the free region of $w(h)$. Therefore, $\sum_i \|\sigma_i\| \geq \bar{\pi}(h)$. Putting all together, we have

$$\begin{aligned} \|\pi(h)\| &= \|\bar{\pi}(h)\| + 2h - z(t) \\ &\leq \sum_{i=1}^k \|\sigma_i\| + \sum_{i=1}^k \|h_i\| \leq 2^{(1-\frac{1}{p})} \sum_{i=1}^k \|s_i\| \end{aligned}$$

which implies that $\|\pi^*\| \leq 2^{(1-\frac{1}{p})} \|\pi_{\text{opt}}\|$. □

By Observation 1, any algorithm that computes an ε -approximation to the optimal VHV-path above a polyhedral terrain T , provides a factor- $(2^{(p-1)/p} + \varepsilon)$ approximation to the L_p -shortest path above T in any L_p metric. The following observation will be a main ingredient of our ε -approximation algorithm in Section 4.

Observation 2 *Let $h > h' > 0$. If $h - h' \leq \Delta/2$ then $\|\pi(h)\| \leq \|\pi(h')\| + \Delta$.*

Proof. Let $L(h) = \|\bar{\pi}(h)\|$. The free region of $w(h)$, $\mathcal{F}(h)$, expands as h increases. Therefore, $L(h)$ is a decreasing function of h . It means that for $h > h'$, $L(h) - L(h') \leq 0$. Using the fact that $\|\pi(h)\| = L(h) + 2h - z(t)$ we get

$$\begin{aligned} \|\pi(h)\| - \|\pi(h')\| &= L(h) - L(h') + 2(h - h') \\ &\leq 2(h - h') \leq \Delta. \end{aligned} \quad \square$$

3 Bounding the Length of the Optimal Path

In this section, we show how to efficiently find a crude approximation to the optimal VHV-path, π^* , above a polyhedral terrain. More precisely, we find a real number that approximates the length of π^* to within a multiplicative-factor of $O(n)$. This crude approximation will be used in the next section to obtain an ε -approximation to π^* .

For $r > 0$, let $C_s(r)$ be a cube of side length $2r$ centered at s . We first prove the following simple lemma.

Lemma 1. *Given a value $r > 0$, we can check in $O(n)$ time whether $C_s(r)$ contains a path from s to t that fully stays on or above T .*

Proof. Let S be the top face of $C_s(r)$ and $w(r)$ be the plane containing S . To see if $C_s(r)$ contains a valid path from s to t , we just need to check if there is a path connecting $s(r)$ to $t(r)$ in $S \setminus T$. (We recall that $s(r)$ and $t(r)$ are the vertical projections of s and t on $w(r)$, respectively). The intersection of T and $w(r)$ forms a set of obstacles $O = \{O_1, O_2, \dots\}$, where each obstacle is a simple polygon (we discard holes inside the obstacles). If T is stored in a proper data structure like a DCEL¹, we can obtain each O_i as a sorted list of its edges in total linear time. For each obstacle O_i , we then compute $S \setminus O_i$ as a set of simple polygons, and use standard point location methods to check if $s(r)$ and $t(r)$ lie in the same polygon. This can be done in time linear to the size of O_i [10]. Thus, performing the check on all obstacles can be done in $O(n)$ overall time. \square

The next lemma provides a lower and an upper bound on the length of π^* .

Lemma 2. *Let π^* be the optimal VHV-path above T . We can find a value r such that $r \leq \|\pi^*\| < 8nr$ in $O(n \log n)$ time and $O(n)$ space.*

Proof. Let r^* be the smallest value for which $C_s(r^*)$ contains a valid path from s to t above T . Clearly, $r^* \leq \|\pi^*\|$. If S is the top face of $C_s(r^*)$, then there is a path from $s(r^*)$ to $t(r^*)$ in $S \setminus T$ that consists of k segments of length at most $4r^*$. It is easy to observe that $k \leq n - 1$. Therefore, $\|\bar{\pi}(r^*)\| \leq 4(n - 1)r^*$, and hence, $\|\pi^*\| \leq \|\bar{\pi}(r^*)\| \leq 4(n - 1)r^* + 2r^* < 4nr^*$.

Now we show how to find a 2-approximation of r^* in $O(n \log n)$ time. Let N be the maximum bit-length of the integers in the input coordinates. It is clear that $r^* \leq 2^N$. Furthermore, we know that the shortest distance between any pair of points in this setting is 2^{-3N} (This is the distance between two parallel planes specified with integer coefficients of bit length at most N , and thus a 3×3 determinant of such integers [9]).

For every integer i , we define $r_i \equiv 2^{i-3N-1}$. It is clear that $r_t \leq r^* \leq r_{t+1}$ for some $t \in [0, 4N]$. We use the idea of binary search to find t using at most $O(\log N)$ queries of the form “if $C_s(r_i)$ contains a valid path from s to t ”. According to Lemma 1, this requires $O(n \log N) = O(n \log n)$ overall time. By setting $r = r_t$, we simply have $r \leq r^* \leq 2r$, and hence $r \leq \|\pi^*\| < 8nr$. \square

¹ Doubly-Connected Edge List [10].

4 The ε -Approximation Algorithm

Let Π be the set of all VHV-paths between s and t that fully stay on or above T . For $R \geq 0$, we denote by Π_R the set of those paths in Π that lie completely in the half-space $z \leq R$. In other words, Π_R is the set of those VHV-paths whose heights are restricted to be at most R . Let π_R^* be the L_p -shortest path in Π_R . For $R < R'$, we have $\Pi_R \subseteq \Pi_{R'} \subseteq \Pi$. Therefore,

$$\|\pi_R^*\| \geq \|\pi_{R'}^*\| \geq \|\pi^*\|$$

where π^* is the optimal path in Π . Furthermore, the following property holds true:

$$\|\pi^*\| \leq R \implies \|\pi_R^*\| = \|\pi^*\|.$$

According to this property, there is a direct correlation between the search parameter R and the length of the optimal path in Π_R . It enables us to use a pseudo approximation framework proposed by Asano *et al.* [1].

For $\varepsilon > 0$, a *pseudo approximation algorithm* for our problem computes a path $\pi(\varepsilon, R) \in \Pi_R$ such that

$$\|\pi(\varepsilon, R)\| \leq \|\pi_R^*\| + \varepsilon R.$$

The following lemma provides an efficient pseudo approximation algorithm for our problem.

Lemma 3. *For $R \geq 0$ and $\varepsilon > 0$, there is a pseudo approximation algorithm that computes a path $\pi(\varepsilon, R) \in \Pi_R$ satisfying $\|\pi(\varepsilon, R)\| \leq \|\pi_R^*\| + \varepsilon R$ in $O(\frac{n}{\varepsilon} \log n)$ time and $O(n \log n)$ space.*

Proof. The algorithm is straightforward: For each $1 \leq i \leq \lceil 2/\varepsilon \rceil$, we compute $\pi(h_i)$ at heights $h_i = i \times \varepsilon R/2$, and then, select the path with the minimum L_p -length among the computed paths as $\pi(\varepsilon, R)$. Let h^* be the maximum z -coordinate of a point on π_R^* , i.e. $\|\pi_R^*\| = \|\pi(h^*)\|$. Clearly, h^* falls in an interval $[h_{k-1}, h_k]$ for some $1 \leq k \leq \lceil 2/\varepsilon \rceil$. Since $h_k - h^* \leq \varepsilon R/2$, Observation 2 implies that $\|\pi(h_k)\| \leq \|\pi(h^*)\| + \varepsilon R$ and hence $\|\pi(\varepsilon, R)\| \leq \|\pi_R^*\| + \varepsilon R$.

For the complexity, we recall that computing each $\pi(h_i)$ is equivalent to constructing a planar L_p -shortest path $\bar{\pi}(h_i)$, which can be accomplished in $O(n \log n)$ time and $O(n \log n)$ space [5]. (Indeed, we need just linear space in L_1 -metric [4]). Computing $\lceil 2/\varepsilon \rceil$ such paths requires $O(\frac{n}{\varepsilon} \log n)$ total time. \square

We call R a *low value* in case $\|\pi(\varepsilon, R)\| \geq R$, and a *high value* otherwise. Asano *et al.* have proved the following nice property.

Lemma 4. [1] *For $\alpha > 0$, if R_l is a low value and R_h is a high value s.t. $R_h \leq \alpha R_l$, then $\|\pi(\varepsilon, R_h)\| < (1 + \alpha \frac{\varepsilon}{1-\varepsilon}) \|\pi^*\|$.*

By assuming $\alpha = 2$ and $\varepsilon \leq 1/2$, we always have $\alpha\varepsilon/(1-\varepsilon) \leq 4\varepsilon$. Using Lemma 4, one can therefore obtain a $(1 + 4\varepsilon)$ -approximation to π^* , for any $\varepsilon \leq$

1/2, by simply finding a low value R_l and a high value R_h such that $R_h \leq 2R_l$. The following algorithm uses this fact to compute an ε -approximate VHV-path.

Algorithm 1 FIND ε -APPROXIMATE VHV-PATH

- 1: Find an r such that $r \leq \|\pi^*\| < 8nr$ (by Lemma 2)
 - 2: Define $R_i \equiv r2^i$, for all $i \geq 0$
 - 3: $l \leftarrow 0$, $h \leftarrow \lceil \log_2 n \rceil + 4$
 - 4: **while** $h - l > 1$ **do**
 - 5: $m \leftarrow \lceil (l + h)/2 \rceil$
 - 6: **if** $R_m \leq \|\pi(\varepsilon, R_m)\|$ **then** $l \leftarrow m$ **else** $h \leftarrow m$
 - 7: Return $\pi(\varepsilon, R_h)$
-

Theorem 1. *Algorithm 1 computes an ε -approximation to the optimal VHV-path above a polyhedral terrain in $O(\frac{n}{\varepsilon} \log n \log \log n)$ time and $O(n \log n)$ space.*

Proof. The correctness of the algorithm easily follows from the following loop invariant: *At the beginning of each iteration, R_l is a low value and R_h is a high value.* Note that before the first iteration, $R_l = r \leq \|\pi^*\| \leq \|\pi(\varepsilon, R_l)\|$ and $R_h \geq 16nr$, thus $\|\pi(\varepsilon, R_h)\| \leq \|\pi_{R_h}^*\| + \varepsilon R_h = \|\pi^*\| + \varepsilon R_h < 8nr + \frac{1}{2}R_h \leq \frac{1}{2}R_h + \frac{1}{2}R_h = R_h$. Upon termination of the loop, we have $R_h = 2R_l$. Therefore, by Lemma 4 the output is a $(1 + 4\varepsilon)$ -factor approximation to π^* for any $\varepsilon \leq 1/2$, and hence, the algorithm can be viewed as a $(1 + \varepsilon')$ -approximation algorithm for any $0 < \varepsilon' \leq 1/8$.

In each iteration of the loop, we need just one call to the pseudo approximation algorithm to verify whether R_m is a low value. The total number of calls to the pseudo approximation algorithm is thus $O(\log \log n)$. It immediately follows from Lemma 3 that the running time of our algorithm is $O(\frac{n}{\varepsilon} \log n \log \log n)$ and its space complexity is $O(n \log n)$. \square

By Observation 1, any ε -approximation to the optimal VHV-path immediately gives a factor- $(2^{(p-1)/p} + \varepsilon)$ approximation to the L_p -shortest path above T . Theorem 1 is therefore equivalent to the following:

Theorem 2. *For any $p \geq 1$, the L_p -shortest path above a polyhedral terrain can be approximated to within a factor of $2^{(p-1)/p} + \varepsilon$ using $O(\frac{n}{\varepsilon} \log n \log \log n)$ time and $O(n \log n)$ space.*

Corollary 1. *For any fixed $p \geq 1$, a 2-approximation to the L_p -shortest path above a polyhedral terrain can be obtained in $O(n \log n \log \log n)$ time and $O(n \log n)$ space.*

Proof. It directly follows from Theorem 2 by picking $\varepsilon = 1/p$, and observing that $1/p \leq 2(1 - 2^{-1/p})$ for all $p \geq 1$. \square

5 Conclusions

In the real world, aircrafts flying over a terrain usually follow a simple pattern: they first fly upwards to a certain height, then travel along a horizontal plane at that height to a point above the target, and finally descend to the target. In this paper, we showed how to efficiently approximate the optimal such vertical-horizontal-vertical path to within a multiplicative factor of $1 + \varepsilon$. This led to a simple and efficient algorithm for approximating the L_p -shortest paths above a polyhedral terrain to within a factor of $2^{(p-1)/p} + \varepsilon$.

While there are several FPTASs for approximating the Euclidean shortest path among a set of polyhedral obstacles, none of these algorithms is specialized for the case where the obstacle is a single polyhedral terrain. An interesting question is whether we can exploit properties of the polyhedral terrains to obtain more efficient FPTASs for this especial case of 3D shortest path problem. The algorithm presented in Section 4 answers this question in affirmative for the L_1 metric. For other L_p metrics ($p \geq 2$) the question remains open.

Acknowledgements. The author would like to thank Timothy M. Chan and Jochen Könemann for their valuable comments.

References

1. T. Asano, D. Kirkpatrick, and C. Yap. Pseudo approximation algorithms with applications to optimal motion planning. *Discrete Comput. Geom.*, 31(1):139–171, 2004.
2. J. F. Canny and J. H. Reif. New lower bound techniques for robot motion planning problems. In *Proc. 28th IEEE Sympos. Found. Comput. Sci.*, pages 49–60, 1987.
3. K. L. Clarkson. Approximation algorithms for shortest path motion planning. In *Proc. 19th ACM Sympos. Theory Comput.*, pages 56–65, 1987.
4. J. Hershberger and S. Suri. An optimal algorithm for Euclidean shortest paths in the plane. *SIAM J. Comput.*, 28(6):2215–2256, 1999.
5. J. S. B. Mitchell. L_1 shortest paths among polygonal obstacles in the plane. *Algorithmica*, 8(1):55–88, 1992.
6. J. S. B. Mitchell. Geometric shortest paths and network optimization. In J.-R. Sack and J. Urrutia, editors, *Handbook of Comput. Geom.*, chapter 15, pages 633–701. Elsevier Science Publishers, 2000.
7. J. S. B. Mitchell. Shortest paths and networks. In J. E. Goodman and J. O’Rourke, editors, *Handbook of Discrete and Comput. Geom.*, chapter 24, pages 445–466. CRC Press, 2nd edition, 2004.
8. J. S. B. Mitchell and M. Sharir. New results on shortest paths in three dimensions. In *Proc. 20th ACM Sympos. Comput. Geom.*, pages 124–133, 2004.
9. C. H. Papadimitriou. An algorithm for shortest-path motion in three dimensions. *Inform. Process. Lett.*, 20(5):259–263, 1985.
10. F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.