Shortest Anisotropic Paths on Terrains *

Mark Lanthier, Anil Maheshwari,
Jörg-Rüdiger Sack

School of Computer Science, Carleton University,
Ottawa, Ontario K1S5B6, Canada

Abstract. We discuss the problem of computing shortest an-isotropic paths on terrains. Anisotropic path costs take into account the length of the path traveled, possibly weighted, and the direction of travel along the faces of the terrain. Considering faces to be weighted has added realism to the study of (pure) Euclidean shortest paths. Parameters such as the varied nature of the terrain, friction, or slope of each face, can be captured via face weights. Anisotropic paths add further realism by taking into consideration the direction of travel on each face thereby e.g., eliminating paths that are too steep for vehicles to travel and preventing the vehicles from turning over. Prior to this work an $O(n^n)$ time algorithm had been presented for computing anisotropic paths. Here we present the first polynomial time approximation algorithm for computing shortest anisotropic paths. Our algorithm is simple to implement and allows for the computation of shortest anisotropic paths within a desired accuracy. Our result addresses the corresponding problem posed in [13].
Keywords: *computational geometry, shortest path, approximation.*

1 Introduction

1.1 Motivation

Shortest path problems arise in many application areas like geographical information systems[5] and robotics. They are among the fundamental problems in computational geometry and other areas such as graph algorithms. In these areas objects are often modeled via terrains. A *terrain* is a set of points and edges (connecting them) whose projection onto the xy-plane forms a triangulation.

A large body of work has centered around the computation of Euclidean shortest paths (we refer the reader to the survey in [10]). For terrains, Sharir and Schorr presented an algorithm for computing Euclidean shortest paths [12] and now we know of a number of different algorithms (see cf. [1, 6, 10]). *Weighted shortest paths* (introduced by [9]) provide more realism in that they can incorporate terrain attributes such as variable costs for different regions. This allows

* Research supported in part by NSERC.

[5] We encountered several shortest path related problems in our R&D on GIS (see [14]); more specifically, e.g., in emergency response time modeling where emergency units are dispatched to emergency sites based on minimum travel times.

one to take into consideration e.g. that the cost of traveling through water, sand, or on a highway is typically different. The NP-hardness and the large time complexities of 3-d shortest paths algorithms even for special problem instances have motivated the search for approximate solutions to the shortest path problem. For weighted shortest path approximations on planar subdivisions or polyhedra, more recently, several algorithms have been proposed [1, 2, 6–8].

In the model introduced by [9], the direction of travel along a face is not captured. The direction of travel plays an important role in determining the physical effects incurred on a vehicle (e.g., car, truck, robot, or even person) traveling along a terrain surface. Through *anisotropism*, we can identify certain directions of travel that represent inclines that are too steep to ascend or unsafe to travel due to possible dangers such as overturning, sliding or wheel slippage. It is for these reasons that we investigate *anisotropic* paths, i.e., paths that take into account the direction of travel as well as length and other physical properties. The model was introduced by Rowe and Ross [11] and it subsumes all previously published criteria for traversal for isotropic weighted region terrain. They present an algorithm which runs in $O(n^n)$ time in the worst case for an n-vertex terrain. The high time complexity of this algorithm motivates the study of approximation algorithms for anisotropic shortest path problems. Refer to [10, 11] for applications, further pointers and discussion on regular grid based heuristics for these problems.

The quality of an approximate solution is assessed in comparison to the correct solution. One particular class of approximation algorithms produces ϵ-approximations of the shortest path. Since mostly the geographic models are approximations of reality anyway and high-quality paths are favored over optimal paths that are "hard" to compute, approximation algorithms are suitable and necessary. In ϵ-approximation algorithms accuracy, arbitrarily high, can be traded off against run-time. Such algorithms are appealing and are thus well studied, in particular, from a theoretical view-point. In this paper we address the problem of computing an ϵ-approximation $\Pi'(s,t)$ to a shortest anisotropic path $\Pi(s,t)$ for a vehicle between two points, s and t, on a terrain $\mathcal{P}$, where $\frac{\|\Pi'(s,t)\|}{\|\Pi(s,t)\|} < 1+\epsilon$, for any given $\epsilon > 0$.

In the case in which a user does not need arbitrary accuracy, we can design a simpler algorithm that produces an approximation within an additive factor of the shortest path. That is, we can compute an approximation $\Pi'(s,t)$ of the weighted shortest anisotropic path $\Pi(s,t)$ between two vertices s and t of G such that $\|\Pi'(s,t)\| \leq \frac{1}{\sin\frac{\alpha}{2}}(\|\Pi(s,t)\| + W|L|)$, where α is the angle that depends upon the feasible directions of travel. Moreover, we can compute this path in $O(nk\log nk + nk^2)$ time, where k is the number of segments of $\|\Pi(s,t)\|$. The details of this simpler algorithm and that of proofs had to be omitted in this paper (due to space restrictions) can be obtained from [7].

1.2 Preliminaries

Our algorithm is designed for the model developed by Rowe and Ross [11]. The model allows two main forces to act against the propulsion of the vehicle, namely friction and gravity. The model assumes no net acceleration over the path from s to t and a cost of zero for turning. Let L be the longest edge of a terrain $\mathcal{P}$. P is composed of n triangular faces, each face $f_j, 1 \leq j \leq n$ having a cost μ_j pertaining to the coefficient of kinetic friction for that face w.r.t. the moving vehicle. Let mg be the weight of the vehicle. Denote by θ the minimum angle between any two adjacent edges of any face on $\mathcal{P}$. Now consider a segment s_i of the shortest path which crosses a face f_j of $\mathcal{P}$. Let ϕ_j be the inclination angle (gradient) of f_j and let φ_i be the inclination angle of s_i w.r.t. to the XY plane (see Figure 1a)). Using this model, the cost of travel for s_i is: $mg(\mu_j \cos\phi_j + \sin\varphi_i) \cdot |s_i|$. We assume that mg is constant for the problem instance and is set to one in our analysis. The cost due to the force of friction is represented by $\mu_j \cos\phi_j \cdot |s_i|$. Therefore, it is convenient to define $w_j = \mu_j \cos\phi_j$ to be the weight of face f_j. Let W (resp. w) be the maximum (resp. minimum) of all $w_j, 1 \leq j \leq n$.

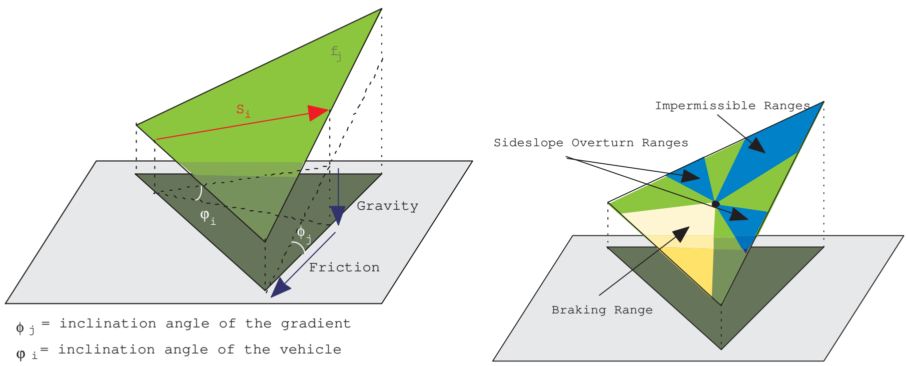

Fig. 1. a) The forces of friction and gravity that act against the propulsion of the vehicle. b) The up to three ranges representing impermissible travel and the braking range of a single face.

The cost due to gravity is represented by $|s_i| \sin\varphi_i$ which is the change in elevation of the path segment. Hence the work expended against gravity is equal to the difference in height between s and t which is independent of the path taken. For certain inclination angles, the cost formula could become negative (i.e., $\varphi_i < -arcsin(\mu_j \cos\phi_j)$) violating the above assumption that there is no net acceleration. The model therefore (in a sense) assumes that the energy gained going downhill is exactly compensated by the energy required to brake. So, vehicles do neither accelerate nor do they gain or lose energy when traveling in a braking range. This is captured by the introduction of *critical braking angles* defined by $\varphi_i = -\arcsin(\mu_j \cos\phi_j)$.

By replacing the $\mu_j \cos\phi_j$ friction factor by $-\sin\varphi_i$ we cancel out the gravity force resulting in zero cost travel. Notice that the negative gravity force has already been extracted from the metric leaving a cost of $-\sin\varphi_i \cdot |s_i|$ for segment s_i. For a braking segment s_i passing through face f_j, $-\sin\varphi_i \geq w_j$.

The model assumes that each face has up to three ranges of angles that define directions on a face that are impermissible for traveling. Together, with the braking range, there are up to four important angular ranges per face as shown in Figure 1b). The boundary angles of the impermissible ranges are called *critical impermissibility angles*. The boundary angles of the braking range are called *critical braking angles*. For the regular angular ranges (i.e., neither impermissible nor braking), the range is bounded by critical impermissibility or braking angles. We will also think of these as the critical angles for the regular range. A path is said to be *valid* if and only if it does not travel in any of the impermissible directions. If any of the impermissible ranges overlap, they are combined to create a single impermissible range. In some cases, the ranges may cover all possible angles and that represents an *isotropic obstacle*.

Given two points s and t on $\mathcal{P}$, there may not be a valid path $\Pi(s,t)$ between s and t on $\mathcal{P}$. The algorithm description and analysis presented here will assume that there exists at least one valid path (i.e. $\Pi(s,t)$) between s and t. Although we make this assumption in the analysis, our algorithm is able to detect the absence of valid paths and report when such a path does not exist.

Let φ_c be a critical impermissibility angle for one of the critical impermissibility ranges of a face f_j of $\mathcal{P}$. Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be the two unit vectors representing the directions on the boundaries of the range. Thus, the angle that $\boldsymbol{u}$ and $\boldsymbol{v}$ make with the horizontal plane is φ_c. Let α_c be the angle between these two vectors when placed end-to-end. Let α_j be the minimum of all α_c for the impermissible ranges and let α be the minimum of all $\alpha_j, 1 \leq j \leq n$. Similarly, we define β as being the minimum of all $\beta_j, 1 \leq j \leq n$, where β_j is the angle between vectors defined by the boundaries of the braking range(s). Let λ to be the minimum angle of all braking and regular range angles.

Rowe and Ross [11] show that this model allows for three types of segments which we denote as *direction types*. We say a segment of a path is *braking* if it travels in a braking heading, otherwise it is a *regular* segment. A path is said to be a *switchback path* if it zig-zags along a matched pair of critical impermissibility directions. In our algorithms, we will treat switchback paths (denoted as z_i) as a single segment, say s_i, and assign a weight to it which incorporates the length of the switchback path itself. It can be shown that a switchback path between two points a and b on a face f_j which uses directions defined by $\boldsymbol{u}$ and $\boldsymbol{v}$ has length at most $\frac{|\overline{ab}|}{\sin(\alpha_j/2)}$.

Let v be a vertex of $\mathcal{P}$. Define h_v to be the minimum distance from v to the boundary of the union of its incident faces. Define a polygonal cap C_v, called a *sphere*, around v, as follows. Let $r_v = \epsilon h_v$ for some $0 < \epsilon$. Let r be the minimum r_v over all v. Let vuw be a triangulated face incident to v. Let u' (w') be at the distance of r_v from v on vu (vw). This defines a triangular sub-face $vu'w'$ of vuw. The spherical cap C_v around v consists of all such sub-faces incident at v.

1.3 Overview of Our Approach

Our approach is to discretize the polyhedral terrain in a natural way, by placing Steiner points along the edges of the polyhedron (as in our earlier subdivision approach [2, 6] but with substantial differences as illustrated below). We construct a graph G containing the Steiner points as vertices and edges as those interconnections between Steiner points that correspond to segments which lie completely in the triangular faces of the polyhedron. The geometric shortest path problem on polyhedra is thus stated as a graph problem so that the existing efficient algorithms (and their implementations) for shortest paths in graphs can be used. The main difference to [6, 2] and to other somewhat related work (e.g., [3–5]) lies in the placement of Steiner points, due to the directional restrictions imposed on the path segments in this case.

We introduce a logarithmic number of Steiner points along each edge of $\mathcal{P}$, and these points are placed in a geometric progression along an edge. They are chosen w.r.t. (i) the vertex joining two edges of a face such that the distance between any two adjacent points on an edge is at most ϵ times the shortest possible path segment that can cross that face between those two points (ii) eight direction ranges (three impermissible, one braking and four regular) such that the approximation segment is of the same type as that of the shortest path segment.

A problem arises when placing these Steiner points near vertices of the face since the shortest possible segment becomes infinitesimal in length. A similar issue was encountered by [1, 2, 5, 4]. The problem arises since the distance between adjacent Steiner points, in the near vicinity of a vertex, would have to be infinitesimal requiring an infinite number of Steiner points. We address this problem by constructing spheres around the vertices which have a very small radius. These spheres help in bounding the number of path segments during the graph construction phase. Note that since switchback paths are allowed, it is possible that the number of segments in a path could be infinite, but there is a concise description of such paths, and that is used in our algorithm. Here lies a further difference to our earlier approach. While the algorithms in [1, 2] never add Steiner points within the spheres centered around each vertex of the polyhedron for anisotropic paths do.

We show that there exist a path in the graph G with cost that is within $(1+f(\epsilon))$ times the shortest path costs, where $f(\epsilon)$ is a function of ϵ and geometric parameters of the terrain. The running time of our algorithm is the cost for computing the graph G plus that of running a shortest path algorithm in G.

2 Our Approximation Scheme

2.1 Computing the Graph

We begin by constructing a graph G_j for each face f_j of $\mathcal{P}$ by adding Steiner points along edges of f_j in three stages. In the *first stage*, we add enough Steiner points to ensure that the distance between adjacent Steiner points on an edge is at most $f(\epsilon)$ times the length of a shortest path segment which passes through it, for some function $f(\epsilon)$ which is independent of n. This is done using the algorithm of [2]. The *second stage* of Steiner points are required to ensure that there exists an approximation segment with the same direction type as a shortest path segment which passes between the same Steiner points. Recall that our model of computation allows for eight direction ranges per face. Second stage adds a set of Steiner points to f_j corresponding to the braking range and (up to four) regular ranges. We give here a description of how to add the Steiner points corresponding to the braking range; the Steiner point sets for the regular ranges are constructed in a similar manner. In the *third stage*, we expand G_j by placing additional vertices within at least one of the spheres around a vertex of f_j.

Stage 1: For each vertex v of face f_j we do the following: Let e_q and e_p be the edges of f_j incident to v. First, place Steiner points on edges e_q and e_p at distance r_v from v; call them q_1 and p_1, respectively. By definition, $|\overline{vq_1}| = |\overline{vp_1}| = r_v$. Define $\delta = (1 + \epsilon \sin\theta_v)$ if $\theta_v < \frac{\pi}{2}$, otherwise $\delta = (1 + \epsilon)$. We now add Steiner points $q_2, q_3, ..., q_{\iota_q - 1}$ along e_q such that $|\overline{vq_j}| = r_v \delta^{j-1}$ where $\iota_q = \log_\delta(|e_q|/r_v)$. Similarly, add Steiner points $p_2, p_3, ..., p_{\iota_p - 1}$ along e_p, where $\iota_p = \log_\delta(|e_p|/r_v)$.

Stage 2: (see Figure 2) Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be the critical angle directions for the range on the plane of f_j. Consider now each vertex v of f_j and apply the following algorithm twice (once as is and then again where $\boldsymbol{u}$ and $\boldsymbol{v}$ are swapped):

$q \leftarrow v$.
WHILE (q does not lie within C_{v_i} of f_j, where $v_i \neq v$) DO {
    $x_q \leftarrow$ the ray from q in direction $\boldsymbol{u}$.
    IF (x_q intersects an edge e of f_j) THEN {
        $p \leftarrow$ intersection point of x_q and e.
        Add p as a Steiner point on e.
        $x_p \leftarrow$ the ray from p in direction $-\boldsymbol{v}$.
        IF (x_p does not intersect an edge e of f_j) THEN STOP
        ELSE {
            $q \leftarrow$ intersection point of x_p with e.
            Add q as a Steiner point on e. } } }

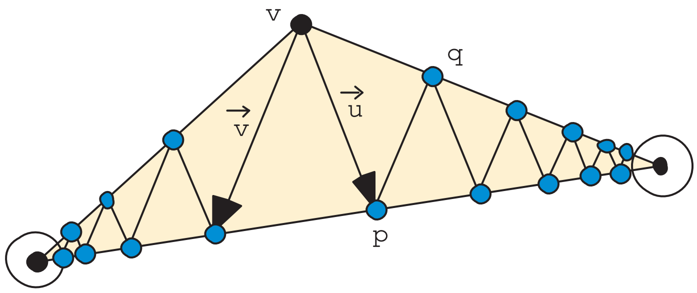

Fig. 2. Adding Steiner points to a face corresponding to a braking range.

Note that when applying this stage to the adjacent faces of f_j, some additional Steiner points will be added to the edges of f_j. When creating the graph for an adjacent face f_{j+1}, each edge of f_j may also gain a set of Steiner points due to this second stage construction. Hence, each edge may have two such sets of Steiner points. The first and second stage of Steiner points along with the vertices of f_j become vertices of G_j. Connect a pair of vertices in G_j by two oppositely directed edges if and only if 1) they represent Steiner points lying on different edges of f_j or 2) they represent adjacent Steiner points lying on the same edge of f_j. In addition, for each vertex of G_j which corresponds to a vertex, say v, of f_j, connect it with two oppositely directed edges to 1) all vertices of G_j that represent Steiner points lying on the edge opposite to v, and 2) the two vertices of G_j corresponding to the two closest Steiner points that lie on the two incident edges of v.

Stage 3: Let q be a Steiner point (or vertex of f_j) on edge e_q of f_j which was added during the first or second stage of Steiner placement (including those from adjacent faces as well). Extend rays from q in the directions of $\boldsymbol{u}$ and $\boldsymbol{v}$. For each ray, if it intersects an edge $e_p \neq e_q$ of f_j at some point x within some C_v of vertex v of f_j then add a Steiner point at x. Add x as a vertex of G_j and add edge $\overrightarrow{qx}$ to G_j. Also add edge $\overrightarrow{xv}$ to G_j. Now extend rays from q in the directions of $-\boldsymbol{u}$ and $-\boldsymbol{v}$. For each ray, if it intersects an edge $e_p \neq e_q$ of f_j at some point x within a distance of r_v of a vertex v of f_j then add a Steiner point at x. Add x as a vertex of G_j and add edge $\overrightarrow{xq}$ to G_j. Also add edge $\overrightarrow{vx}$ to G_j. Let q_a and q_{a+1} be two adjacent Steiner points added on an edge within a sphere C_v as just mentioned. Let p_b and p_{b+1} be the Steiner points that generated q_a and q_{a+1}, respectively. If $|\overline{q_a q_{a+1}}| > r_v(\delta - 1)$ then we add additional evenly spaced Steiner points between q_a and q_{a+1}. We add only enough Steiner points to ensure that the distance between two adjacent points is at most $r_v(\delta - 1)$. Once again, connect each of these new Steiner points, say p to v with two oppositely directed edges. Also, connect p to p_b and p_{b+1} with two oppositely directed edges each. Keep in mind that although we just described the addition of these Steiner points with respect to the two critical directions $\boldsymbol{u}$ and $\boldsymbol{v}$ for the braking range, we must also add similar sets of Steiner points for the regular ranges.

Having added vertices and edges to G_j we must now assign appropriate weights to the edges. For each edge $\overrightarrow{ab}$ of G_j, we set its weight as follows: If $\overrightarrow{ab}$ is regular then its weight is set to $w_j|\overline{ab}|$. If $\overrightarrow{ab}$ is braking then its weight is set to $-\sin\theta_i|\overline{ab}|$ where θ_i is the declination angle of $\overrightarrow{ab}$. If $\overrightarrow{ab}$ is switchback then its weight is set to $\frac{w_j}{\sin\frac{\alpha}{2}}|\overline{ab}|$. This completes the construction of G_j. The graph G is defined to be the union $G_1 \cup G_2 \cup ...G_n$.

Claim 2.11. *G is connected and has $O(n\log_\delta(|L|/r) + n\log_{\mathcal{F}}(r/|L|))$ vertices and $O(n(\log_\delta(|L|/r) + \log_{\mathcal{F}}(r/|L|))^2)$ edges, where $\mathcal{F} = \frac{1+\cos(\theta+\lambda)}{1+\cos(\theta-\lambda)}$, θ is the minimum angle between any two adjacent edges of any face on $\mathcal{P}$, and λ is the minimum of all braking and regular range angles.*

2.2 Constructing the Approximated Path

We describe here the construction of a path $\Pi'(s,t)$ in G. In the section to follow, we will bound the cost of this path. Note however that Dijkstra's algorithm may produce a better path than the one constructed here. Recall that a switchback path z_i of $\Pi(s,t)$ within face f_j is represented with a single segment (i.e. s_i) of $\Pi(s,t)$ whose weight encapsulates the distance of the switchback path. Each s_i, must be of one of the following types:

i) $s_i \cap C_v = \emptyset$,

ii) $s_i \cap C_v =$ subsegment of s_i, or

iii) $s_i \cap C_v = s_i$.

Let $C_{\sigma_1}, C_{\sigma_2}, ..., C_{\sigma_\kappa}$ be a sequence of spheres (listed in order from s to t) intersected by type ii) segments of $\Pi(s,t)$ such that $C_{\sigma_a} \neq C_{\sigma_{a+1}}$. Now define subpaths of $\Pi(s,t)$ as being one of two kinds:

Definition 1. Between-sphere subpath: *A path consisting of a type ii) segment followed by zero or more consecutive type i) segments followed by a type ii) segment. These subpaths will be denoted as $\Pi(\sigma_a, \sigma_{a+1})$ whose first and last segments intersect C_{σ_a} and $C_{\sigma_{a+1}}$, respectively. We will also consider paths that begin or/and end at a vertex to be a degenerate case of this type of path containing only type i) segments.*

Definition 2. Inside-sphere subpath: *A path consisting of one or more consecutive type iii) segments all lying within the same C_{σ_a}; these are denoted as $\Pi(\sigma_a)$. (Note that inside-sphere subpaths of $\Pi(s,t)$ always lie between two between-sphere subpaths. That is, $\Pi(\sigma_a)$ lies between $\Pi(\sigma_{a-1}, \sigma_a)$ and $\Pi(\sigma_a, \sigma_{a+1})$).*

Let x and y be the endpoints of s_i and let x (respectively y) lie on edge e_q (respectively e_p) of f_j. Let q_a and q_b (respectively p_a and p_b) be the Steiner points on e_q (respectively e_p) between which x (respectively y) lies.

Claim 2.21. *At least one of $\overrightarrow{q_a p_b}$ or $\overrightarrow{q_b p_a}$ is of the same direction type as s_i.*

We begin our path construction by choosing a segment s'_i in G_j which approximates a segment s_i crossing face f_j. If s_i is a type i) or type ii) segment, then choose s'_i to be one of $\overrightarrow{q_a p_a}$, $\overrightarrow{q_a p_b}$, $\overrightarrow{q_b p_a}$ and $\overrightarrow{q_b p_b}$ such that s'_i is of the same direction type as s_i and $\|s'_i\|$ is minimized. Claim 2.21 ensures that at least one of these segments is of the same type as s_i. For the sake of analysis, we will assume that s'_i is chosen so as to have the same direction type as s_i and we will bound s'_i accordingly. In practice however, we may choose a segment with less cost, since we are choosing the minimum of these four. Note that this choice also pertains to the special case in which $e_q = e_p$. Note also that if s_i is of type ii), then one of q_a, q_b, p_a or p_b may degenerate to a vertex of f_j. In the case where s_i is a type iii) segment, there is no corresponding segment s'_i in $\Pi'(s,t)$.

At this point, we have approximations for all type i) and type ii) segments but they are disconnected and therefore do not form a path joining s and t. We will now add edges joining consecutive type i) or type ii) segments of $\Pi(s,t)$. Let s_i and s_{i+1} be two consecutive segments of $\Pi(s,t)$ that are type i) or type ii) with corresponding approximation segments s'_i of G_j and s'_{i+1} of G_{j+1}, respectively. Let e be the edge of $\mathcal{P}$ joining faces f_j and f_{j+1}. Let q be the endpoint of s'_i lying on e and let p be the endpoint of s'_{i+1} lying on e. It is easily seen that either $q = p$ or q and p are adjacent Steiner points on e. If $q = p$, then s'_i and s'_{i+1} are already connected. If $q \neq p$ then let s''_i be the edge in G_j from q to p. s''_i is used to

The addition of these segments (i.e. all s''_i) ensures that all segments of between-sphere subpaths are connected to form subpaths. We now need to interconnect the between-sphere subpaths so that $\Pi'(s,t)$ is connected.

Consider two consecutive between-sphere subpaths of $\Pi'(s,t)$, say $\Pi'(\sigma_{a-1},\sigma_a)$ and $\Pi'(\sigma_a,\sigma_{a+1})$. They are disjoint from one another, however, the first path ends at a Steiner point within sphere C_{σ_a} and the second path starts at a Steiner point within C_{σ_a}. Join the end of $\Pi'(\sigma_{a-1},\sigma_a)$ and the start of $\Pi'(\sigma_a,\sigma_{a+1})$ to vertex v_{σ_a} by two segments (which are edges of G created in Stage 3). These two segments together will form an inside-sphere subpath and will be denoted as $\Pi'(\sigma_a)$. This step is repeated for each consecutive pair of between-sphere subpaths so that all subpaths are joined to form $\Pi'(s,t)$. Constructing a path in this manner results in a connected path that lies on the surface of $\mathcal{P}$.

2.3 Bounding the Approximation

We give a bound $\|\Pi'(s,t)\|$ on the cost of $\Pi'(s,t)$. To begin, a bound is shown for each of the between-sphere path segments. The claims to follow bound the approximation segments of the type i) and type ii) face crossing segments of $\Pi(s,t)$. Assume therefore that s'_i is a type i) or type ii) face-crossing segment. The claims give bounds for the three possible direction types of s'_i. That is, we bound the weighted cost of s'_i for the cases in which s'_i (and hence s_i) is regular, braking and switchback, respectively. For the following claims, we will assume that $s'_i = \overline{q_a p_b}$; similar proofs hold when $s'_i = \overline{q_b p_a}$.

Claim 2.31. *Let s_i and s'_i be two segments as discussed above, passing through a face f_j which has weight w_j. Then*
i) if s_i and s'_i are regular then $\|s'_i\| \leq (1+2\epsilon)\|s_i\|$
ii) if s_i and s'_i are braking then $\|s'_i\| \leq \left(1+\frac{2\epsilon}{w_j}\right)\|s_i\|$.
iii) if s_i and s'_i are switchback then $\|s'_i\| \leq \left(1+\frac{2\epsilon}{\sin\frac{\alpha}{2}}\right)\|s_i\|$
iv) $\|s''_i\| \leq \frac{\epsilon}{\sin\frac{\alpha}{2}}\|s_i\|$.

Lemma 1. *If $\Pi'(\sigma_{a-1},\sigma_a)$ is a between-sphere subpath of $\Pi'(s,t)$ corresponding to an approximation of $\Pi(\sigma_{a-1},\sigma_a)$ then $\|\Pi'(\sigma_{a-1},\sigma_a)\| \leq (1+\max(\frac{1}{\sin\frac{\alpha}{2}} + \frac{2}{w}, \frac{3}{\sin\frac{\alpha}{2}})\epsilon)\|\Pi(\sigma_{a-1},\sigma_a)\|$, where w is the minimum weight of the faces of $\mathcal{P}$.*

Proof Sketch: Let s'_i be a segment of $\Pi'(\sigma_{a-1},\sigma_a)$ which approximates a segment s_i of $\Pi(\sigma_{a-1},\sigma_a)$ passing through face f_j. By Claim 2.31 we have $\|s'_i\| \leq (1+2\epsilon\max(\frac{1}{w},\frac{1}{\sin\frac{\alpha}{2}}))\|s_i\|$. We can charge the cost of each s''_i to s'_i. Therefore, we can say from Claim 2.31(iv) that $\|s'_i\| + \|s''_i\| \leq (1+\max(\frac{1}{\sin\frac{\alpha}{2}} + \frac{2}{w}, \frac{3}{\sin\frac{\alpha}{2}})\epsilon)\|s_i\|$.

Hence, each segment s_i' of $\Pi'(\sigma_{a-1},\sigma_a)$ has cost at most $(1+\max(\frac{1}{\sin\frac{\alpha}{2}}+\frac{2}{w},\frac{3}{\sin\frac{\alpha}{2}})\epsilon)\|s_i\|$ and $\|\Pi'(\sigma_{a-1},\sigma_a)\| \leq (1+\max(\frac{1}{\sin\frac{\alpha}{2}}+\frac{2}{w},\frac{3}{\sin\frac{\alpha}{2}})\epsilon)\|\Pi(\sigma_{a-1},\sigma_a)\|$.

Claim 2.32. *Let $\Pi'(\sigma_{a-1},\sigma_a)$ be a between-sphere subpath of $\Pi'(s,t)$ corresponding to an approximation of $\Pi(\sigma_{a-1},\sigma_a)$ then*
$\|\Pi'(\sigma_a)\| \leq \left(\frac{2W\epsilon}{w(1-2\epsilon)\sin\frac{\alpha}{2}}\right)\|\Pi(\sigma_{a-1},\sigma_a)\|$, *where* $0<\epsilon<\frac{1}{2}$.

Proof sketch: The distance between $C_{\sigma_{a-1}}$ and C_{σ_a} must be at least $(1-2\epsilon)h_{v_{\sigma_a}}$. Since $\Pi(\sigma_{a-1},\sigma_a)$ is a between-sphere subpath, it intersects both $C_{\sigma_{a-1}}$ and C_{σ_a}. Thus $|\Pi(\sigma_{a-1},\sigma_a)| \geq (1-2\epsilon)h_{v_{\sigma_a}}$. By definition, $\Pi'(\sigma_a)$ consists of exactly two segments which together have length satisfying $|\Pi'(\sigma_a)| \leq 2r_{v_{\sigma_a}} = 2\epsilon h_{v_{\sigma_a}}$. Thus, $|\Pi'(\sigma_a)| \leq \frac{2\epsilon}{1-2\epsilon}|\Pi(\sigma_{a-1},\sigma_a)|$. In the worst case, we can assume that segments of $\Pi'(\sigma_a)$ are impermissible and that $\Pi(\sigma_{a-1},\sigma_a)$ is braking. Hence, $\|\Pi'(\sigma_a)\| \leq \frac{W}{\sin\frac{\alpha}{2}}|\Pi'(\sigma_a)| \leq \frac{2W\epsilon}{w(1-2\epsilon)\sin\frac{\alpha}{2}}\|\Pi(\sigma_{a-1},\sigma_a)\|$.

Lemma 2. *If $\Pi(s,p)$ is a shortest anisotropic path in $\mathcal{P}$, where s is a vertex of $\mathcal{P}$ and p is a vertex of G then there exists an approximated path $\Pi'(s,p)\in G$ such that $\|\Pi'(s,p)\| \leq (1+f(\epsilon))\|\Pi(s,p)\|$ where $0<\epsilon<\frac{1}{2}$ and*
$f(\epsilon) = \left(\epsilon\left(\frac{2W\epsilon}{w(1-2\epsilon)\sin\frac{\alpha}{2}}+\max\left(\frac{1}{\sin\frac{\alpha}{2}}+\frac{2}{w},\frac{3}{\sin\frac{\alpha}{2}}\right)\right)\right)$.

Proof sketch: Using the results of Claim 2.32 and Lemma 1, it can be shown that $\|\Pi'(\sigma_{a-1},\sigma_a)\|+\|\Pi'(\sigma_a)\| \leq \left(1+\epsilon\left(\frac{2W\epsilon}{w(1-2\epsilon)\sin\frac{\alpha}{2}}+\max\left(\frac{1}{\sin\frac{\alpha}{2}}+\frac{2}{w},\frac{3}{\sin\frac{\alpha}{2}}\right)\right)\right)\|\Pi(\sigma_{a-1},\sigma_a)\|$. This essentially "charges" the length of an inside-sphere subpath to a between-sphere subpath. The union of all such subpaths form $\Pi'(s,p)$. This allows us to approximate $\Pi'(s,p)$ within the bound of $1+\epsilon\left(\frac{2W\epsilon}{w(1-2\epsilon)\sin\frac{\alpha}{2}}+\max\left(\frac{1}{\sin\frac{\alpha}{2}}+\frac{2}{w},\frac{3}{\sin\frac{\alpha}{2}}\right)\right)$ times the total cost of all the between-sphere subpaths of $\Pi(s,p)$. Since $\Pi(s,p)$ has cost at least that of its between-sphere subpaths, the lemma holds true.

Theorem 1. *Let $\mathcal{P}$ be a polyhedral surface with maximum and minimum face weights W and w, respectively such that $W \geq \sin\frac{\alpha}{2}$, where α is the minimum angle defined by any pair of matched critical impermissible and braking angles. Let $\Pi(s,t)$ be a shortest weighted path on $\mathcal{P}$, where s and t are vertices of $\mathcal{P}$ then there exists an approximated path $\Pi'(s,t)\in G$ such that $\|\Pi'(s,t)\| \leq (1+f(\epsilon))\|\Pi(s,t)\|$, where $f(\epsilon)=\left(\frac{3W\epsilon}{w\sin\frac{\alpha}{2}}\right)$. Moreover, $\|\Pi'(s,t)\|$ can be computed by running Dijkstra's shortest path algorithm on the graph computed in Claim 2.11.*

3 Conclusion

We presented an algorithm for computing an ϵ-approximation to a shortest anisotropic path on the terrain. A similar, but simplified (therefore omitted here) methodology allows for the computation of an approximation to within an additive factor of the shortest anisotropic path [7]. Both algorithms expand on and generalize edge subdivision schemes we had introduced earlier [1, 2]. Thus one general technique gives rise to Euclidean, weighted and anisotropic path algorithms. The differences are not so significant for the implementation as they are for the analysis. All graph construction schemes are easy to implement and then require only running a shortest path algorithm in a graph. We believe that the approximations within an additive factor will be of special interest for practitioners. The ϵ-approximations are also of theoretical interest as they require new ideas (as also discussed here).

Acknowledgments: The authors would like to thank Lyudmil Aleksandrov for helpful discussion on Claim 2.11.

References

1. L. Aleksandrov, M. Lanthier, A. Maheshwari and J.-R. Sack, "An ϵ-Approximation Algorithm for Weighted Shortest Path Queries on Polyhedral Surfaces", *14th European Workshop on Computational Geometry*, Barcelona, Spain, 1998.
2. L. Aleksandrov, M. Lanthier, A. Maheshwari and J.-R. Sack, "An ϵ-Approximation Algorithm for Weighted Shortest Paths on Polyhedral Surfaces", *SWAT '98*, Stockholm, Sweden, 1998.
3. J. Choi, J. Sellen and C.K. Yap, "Approximate Euclidean Shortest Path in 3-Space", *Proc. 10th Annual Symp. on Computational Geometry*, 1994, pp. 41-48.
4. G. Das and G. Narasimhan, "Short Cuts in Higher Dimensional Space", *Proceedings of the 7th Annual Canadian Conference on Computational Geometry*, Québec City, Québec, 1995, pp. 103-108.
5. C. Kenyon and R. Kenyon, "How To Take Short Cuts", *Discrete and Computational Geometry*, Vol. 8, No. 3, 1992, pp. 251-264.
6. M. Lanthier, A. Maheshwari and J.-R. Sack, "Approximating Weighted Shortest Paths on Polyhedral Surfaces", *Proceedings of the 13th Annual ACM Symposium on Computational Geometry*, 1997, pp. 274-283.
7. M. Lanthier, "Shortest Path Problems on Polyhedral Surfaces", *Ph.D. Thesis in progress*, School of Computer Science, Carleton University, Ottawa, Canada, 1999, available at *http://www.scs.carleton.ca/~ lanthier/research/papers.html*
8. C. Mata and J. Mitchell, "A New Algorithm for Computing Shortest Paths in Weighted Planar Subdivisions", *Proceedings of the 13th Annual ACM Symposium on Computational Geometry*, 1997, pp. 264-273.
9. J.S.B. Mitchell and C.H. Papadimitriou, "The Weighted Region Problem: Finding Shortest Paths Through a Weighted Planar Subdivision", *Journal of the ACM*, **38**, January 1991, pp. 18-73.
10. J.S.B. Mitchell, "Geometric Shortest Paths and Network Optimization", in J.-R. Sack and J. Urrutia Eds.,*Handbook on Computational Geometry*, Elsevier Science B.V., to appear 1998.
11. N.C. Rowe, and R.S. Ross, "Optimal Grid-Free Path Planning Across Arbitrarily Contoured Terrain with Anisotropic Friction and Gravity Effects", IEEE Transactions on Robotics and Automation, Vol. 6, No. 5, October 1990, pp. 540-553.
12. M. Sharir and A. Schorr, "On Shortest Paths in Polyhedral Spaces", *SIAM Journal of Computing*, **15**, 1986, pp. 193-215.
13. R. Tamassia et al., "Strategic Directions in Computational Geometry", ACM Computing Surveys, Vol. 28, No. 4, December 1996.
14. Paradigm Group Webpage, School of Computer Science, Carleton University, http://www.scs.carleton.ca/~ gis.