Approximation algorithms for geometric shortest path problems*

Lyudmil Aleksandrov† Anil Maheshwari ‡ Jörg-Rüdiger Sack §

October 13, 1999

Abstract

We consider the classical geometric problem of determining a shortest path through a weighted domain. We present approximation algorithms that compute ε-*short* paths, i.e., paths whose costs are within a factor of $1+\varepsilon$ of the shortest path costs, for an arbitrary constant $\varepsilon > 0$, for the following geometric configurations:

SPPS Problem: We are given a polyhedron $\mathcal{P}$ consisting of n convex faces[1] and each face has a positive non-zero real valued weight. The *shortest path on polyhedral surface problem* (SPPS) is to compute a path of least cost that remains on the surface of $\mathcal{P}$ between any two vertices, where the cost of the path is defined to be the weighted sum of Euclidean lengths of the sub-paths within each face. Our algorithm runs in $O(\frac{n}{\varepsilon}\log\frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}}+\log n))$ time for $\varepsilon < 1$. The run time improves to $O(n\log n)$ for $\varepsilon \geq 1$ and to $O(\frac{n}{\varepsilon}\log\frac{1}{\varepsilon}\log n)$ when all weights are equal.

WRP-3D Problem: We are given a subdivision of $\Re^3$ consisting of n convex regions. Each face has associated with it a positive non-zero real valued weight. The *shortest path problem in three dimensions* (SP3D) is to compute a path of least cost between any two vertices, where the cost of the path is defined to be the weighted sum of Euclidean lengths of the sub-paths within each region. We present the first polynomial time approximation scheme. Our algorithm runs in $O(\frac{n}{\varepsilon^3}\log\frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}}+\log n))$ time. The run time improves to $O(\frac{n}{\varepsilon^3}\log\frac{1}{\varepsilon}\log n))$ when all weights are equal.

1 Introduction:

1.1 Motivation: Shortest path problems are among the fundamental problems studied in computational geometry and other areas including graph algorithms, geographical information systems (GIS) and robotics. Existing algorithms for most of the interesting shortest path problems are either very complex and/or have very large time and space complexities. Hence they are unappealing to practitioners and pose a challenge to theoreticians. This coupled with the fact that 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.

1.2 Related Work: Shortest path problems in computational geometry can be categorized by various factors which include the dimensionality of space, the type and number of objects or obstacles (e.g., polygonal obstacles, convex or non-convex polyhedra, ...), and the distance measure used (e.g., Euclidean, number of links, or weighted distances). Several research articles, including surveys (see [26, 29, 30]), have been written presenting the state-of-the-art in this active field. Here we discuss those contributions which relate more directly to our work; these are in particular stated in 2 and 3-dimensional weighted scenarios.

In two dimensions several variations of shortest path problems have been studied over the last two decades. This includes computing Euclidean shortest paths and answering shortest path queries between two points inside a simple polygon and amidst polygonal obstacles. Of particular interest is the *weighted region problem* (abbreviated as SPPS here) introduced in [28]; it is a natural generalization of the shortest path problem in polygonal domains. In their version [28] of the SPPS problem a planar triangulated subdivision is given consisting of n faces, where each face has a positive non-zero weight. Using an algorithm based on the "continuous Dijkstra's method" Mitchell and Papadimitriou [28] provide an approximation algorithm to compute a (weighted) ε-short path; it runs in $O(n^8\log(nNW/w\epsilon))$ time using $O(n^4)$ space, where N is the largest integer coordinate of any

*Research supported by NSERC and NCE GEOIDE

†Bulgarian Academy of Sciences. lyualeks@bgearn.acad.bg

‡School of Comp. Sci., Carleton U. maheshwa@scs.carleton.ca

§School of Comp. Sci., Carleton U. sack@scs.carleton.ca

[1] For ease of notation we assume throughout the SPPS portion of this paper that the number of faces is proportional to the number of vertices although our algorithms are stated in the general sense.

vertex of the triangulation and W (w) is the maximum (minimum) weight of any face of the triangulation.

Lanthier et al. [23] have described several algorithms for the SPPS problem; the cost of the approximation is no more than the shortest path cost plus an (additive) factor of $W|LongestEdge|$, where $LongestEdge$ is the longest edge, W is the largest weight among all faces. As their experimental analysis shows these algorithms are also of practical value. They [25] also developed approximation algorithms for anisotropic shortest paths problems. Mata and Mitchell [27] have proposed an algorithm that constructs a graph (pathnet) which can be searched to obtain an approximate path and it runs in $O(kn^3)$ time, where k depends upon ε, the weights and the geometric parameters. Aleksandrov et al. [3] propose ε-algorithms to solve the SPPS problem in $O((\frac{n}{\varepsilon^2} + \frac{n\log n}{\varepsilon})\log\frac{1}{\varepsilon})$ time.

Given a set of pairwise disjoint polyhedra in $\Re^3$ and two points s and t, the problem of computing a shortest path between s and t that avoids the interiors of polyhedra is $NP-Hard$ [6]. The shortest path problem amidst (disjoint) convex polyhedra can be solved in time exponential in the number of polyhedral objects as was shown by Sharir [32]. For two polyhedral obstacles with a total of n vertices, Baltsan and Sharir [5] presented an $O(n^3\log n)$ time shortest path algorithm. 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 shortest path problems.

Papadimitriou [31] was the first to study fully polynomial time approximation algorithms for the *Euclidean shortest path problem in three dimensions* (termed as ESP3D), where the interiors of the obstacles have infinite weight and the rest of the space has a uniform finite weight. Choi et al. [10, 11] present a refinement of the scheme proposed by [31] which provide an ε-approximation for ESP3D; their algorithm runs in $O((n^3M\log M + (nM)^2).\mu(W))$ time, where $W = O(\log(\frac{n}{\varepsilon}) + L)$, $M = O(nL/\varepsilon)$ and each vertex of an obstacle is specified by an L-bit integer. Both of these algorithms divide the edges of the polyhedra into intervals by introducing Steiner points based on a geometric progression which depends on ε and the location of the source vertex, and then build a graph on these Steiner points. The ESP3D problem is solved by computing a shortest path in the graph using Dijkstra's algorithm [15]. Clarkson [8] provides a solution for the ESP3D problem. His algorithm produces an ε-short path and runs in $O(n^2\lambda(n)\log(n/\varepsilon)/(\varepsilon^4) + n^2\log n\rho\log(n\log\rho))$ time, where ρ is the ratio of the longest obstacle edge to the distance between the source and the target vertex, $\lambda(n) = \alpha(n)^{O(\alpha(n))^{O(1)}}$, $\alpha(n)$ is the inverse Ackermann's function.

Recently, there have been several results on approximation algorithms for computing Euclidean shortest paths on (convex) polyhedral surfaces. Hershberger and Suri [20] present a simple linear time algorithm that computes a 2-approximation to the shortest path. Har-Peled et al. [17] extend this result to provide an algorithm to compute an ε-approximation of the shortest path; it runs in $O(n\min\{1/\varepsilon^{1.5}, \log n\} + 1/\varepsilon^{4.5}\log(1/\varepsilon))$ time. Agarwal et al. [2] provide an improved algorithm that runs in $O(n\log\frac{1}{\varepsilon} + \frac{1}{\varepsilon^3})$ time. Har-Peled [18] provides an algorithm to solve a variation of this problem which computes, in $O(\frac{\log n}{\varepsilon^{1.5}} + \frac{1}{\varepsilon^3})$ time, an ε-approximation for a path between two points on the surface of a convex polytope, after a linear time preprocessing. All of these algorithms crucially exploit the properties of convex polyhedra. Varadarajan and Agarwal [1] provide an algorithm that computes a path on a, possibly non-convex, polyhedron that is at most $7(1+\varepsilon)$ times the shortest path length; it runs in $O(n^{5/3}\log^{5/3} n)$ time. They also present a slightly faster algorithm that returns a path which is at most $15(1+\varepsilon)$ times the shortest path length. Moreover, the algorithms of [23, 27] apply for Euclidean shortest path problems on polyhedral surfaces as well. We note that the Chen and Han's algorithm [7] can compute an exact Euclidean shortest path between two points on a polyhedral surface in $O(n^2)$ time.

1.3 Summary of our contributions:

Our results improve upon previous results in a variety of ways:

1. *SPPS problem:* Considering the real-RAM model, we improve upon the result of Mitchell and Papadimitriou [28], by a factor of about n^7 in the problem size n. More specifically, our algorithm runs in $O(\frac{n}{\varepsilon}\log\frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}} + \log n))$ time, whereas Mitchell and Papadimitriou's algorithm runs in $O(n^8\log(nNW/w\epsilon))$ time. Although the estimates include some geometric parameters, our result improves on that of [28] if $1/\varepsilon^{1.5}$ is less than $n^7/\log n$ and on that of [27] if $n^2 > \varepsilon^{-\frac{1}{2}}\log\frac{1}{\varepsilon}$ (which holds for most realistic problem instances). Although the improvement in the time complexity over [3] by a factor of $\frac{1}{\sqrt{\varepsilon}}$ may not look substantial, note that the margin to maneuver is very small, the constants arising out of geometric parameters is substantially better and the techniques developed in the process are novel and also we show that they are helpful in solving the three dimensional problem SP3D, which none of the previous techniques could.

2. *Unweighted SPPS problem:* We improve upon the results of Varadarajan and Agarwal [1] in the worst case time complexity as well as in terms of the approximation factors. They provide $7(1+\varepsilon)$ and

$15(1+\varepsilon)$ approximations whereas ours is a true, i.e., $(1+\varepsilon)$-approximation. For $\varepsilon > 1$ our algorithm runs in $O(n \log n)$ time, whereas theirs takes $O(n^{5/3} \log^{5/3} n)$ time.

3. *SP3D Problem:* We provide the first polynomial time ε-approximation scheme.

4. *Unweighted SP3D Problem:* For the unweighted version, namely the ESP3D problem, our worst case time complexities are better than those of [31, 10, 11] by at least a factor of n. Moreover, similar to the SPPS problem, our algorithm is simple and works for any pair of source and target vertex.

5. *Geometric parameters:* The constants that are hidden by the big-O notation may play an important role. Our analysis (for SPPS) reveals fairly precisely the constants of the geometric portion of our algorithm (see e.g., Lemma 2.3). These constants depend on the geometric constellation of the given problem instance. The constants in previous approaches depend on ratios of maximum weight to minimum weight, maximum length v/s minimum length of edges, minimum angle of a face, etc. All our constants are averages over all faces and the dependence on the weights is square-root of the ratio of the maximum weight to the minimum weight. This in itself is significant, although it does not show up in the asymptotic analysis.

6. *Simplicity:* Our algorithms are conceptually simple (in particular, compared to [28]), use few and standard geometric primitives and employ an easily implemented modification of Dijkstra's algorithm.

7. *Practicality potential:* Variants of the algorithm proposed here (for the SPPS), where the approximation factor is an additive factor, have been implemented and their practicality has been demonstrated (see [23]).

8. *Source independence:* Most of the previous approaches, including [28, 1], build a graph over the surface, by using the knowledge of the location of the source vertex. This requires rebuilding of the graph if the location of the source vertex changes. Our approach does not require the knowledge of the location of the source or target vertex for building the graph, and hence one graph serves for any pair of source and target vertex.

To solve these problems, we employ the traditional technique of partitioning a continuous geometric search space into discrete combinatorial search space by designing an appropriate mesh. Simplifying, the general strategy taken here is to place Steiner points on edges and inside faces (in 3-D), interconnect the Steiner points within each face and then show that an ε-short path exists approximating any (true) shortest path.

The Steiner point placement is novel (e.g., [31] requires knowledge of the source point and thus needs to be recomputed for every different choice of source point). One of the problems that arise is to place Steiner points near vertices and near edges in 3-D. As, near vertices (edges in 3-D), the distance between adjacent path vertices can become arbitrarily small, an infinite number of Steiner points would be required for the approximation (see also [22, 13]). Here, we address this problem by constructing "spheres" around the vertices and "spindles" around edges of the faces; ensuring that Steiner points are placed outside these regions. (Finding the right radius is non-trivial.) This allows us to put a lower bound on the length of the smallest possible edge that passes between two adjacent Steiner points and hence we are able to add a finite number of Steiner points. Now the challenge is to still show that an ε-approximation scheme is achievable.

Another issue that needs attention is that the constructed graph has considerable large size and it would be inappropriate to search it and hoping for the desired time complexities. We reduce the size of the search space during an execution of Dijkstra's algorithm by deriving geometric properties of Snell's law of refraction for a discrete domain. Snell's law is typically stated for continuous domains. We also describe a variant of Dijkstra's algorithm in which the execution is restricted to a sparse set of potential edges, given that the preceding edge on a path is known. Employing both together and using geometric spanners we are able to construct a path of desired accuracy within the stated complexity bounds.

2 SPPS Problem

2.1 Model: Let $\mathcal{P}$ be a polyhedral surface, whose faces are convex polygons. Let m be the number of edges and n be the number of faces of $\mathcal{P}$. Non-zero positive weights $w_1, \ldots, w_n$ are associated with faces $f_1, \ldots, f_n$ representing costs of traveling along them. The cost of traveling along an edge of $\mathcal{P}$, or otherwise the weight associated with an edge is the minimum of the weights of the two neighboring faces. The cost of a path Π on $\mathcal{P}$ is defined by $\|\Pi\| = \sum_{i=1}^{n} w_i |\Pi_i|$, where $|\Pi_i|$ denotes the Euclidean length of the intersection $\Pi_i = \Pi \cap f_i$. (Edges are assumed to be part of the face from which they inherit their weight.)

Given two distinct points u and v on $\mathcal{P}$ a minimum cost path $\Pi(u,v)$ joining u and v is called *geodesic path*. In this setting, it is well known that geodesic paths are simple (non self-intersecting) and consist of a sequence of segments, whose endpoints are on the edges of $\mathcal{P}$. More precisely, each segment is of one of the following two types: 1) *face-crossing*: a segment which crosses a face joining two points on its boundary that lie on different edges 2) *edge-using*: a segment which (partially) follows an edge.

We define *linear paths* to be paths consisting of face-crossing and edge-using segments exclusively. A linear path $\Pi(u,v)$ is represented as a sequence of its segments $\{s_1,\ldots,s_{l+1}\}$ or/and as sequence of points $a_0,\ldots,a_{l+1}$ lying on adjacent edges of $\mathcal{P}$, that are endpoints of these segments, i.e., $s_i=(a_{i-1},a_i)$, $u=a_0$, and $v=a_{l+1}$. Points a_i that are not vertices of $\mathcal{P}$ are called *bending points* of the path. The local behavior of a linear path around a bending point is described by two oriented angles, which we call *in-angle* and *out-angle* at the considered bending point.

Precisely, let a be a bending point on a linear path Π lying on an edge e. Let s' be the segment preceding a on Π. Then we define the in-angle $\varphi=\varphi(a,\Pi)$ to be the acute clockwise angle between the normal to e at a and s'. Similarly, the out-angle ψ is the acute clockwise angle between the normal to e and the segment s'' succeeding a on Π. Note that in-angles and out-angles are always in $[-\pi/2,\pi/2]$. If the absolute value of an in-angle at some bending point a is $\pi/2$ then the segment preceding a is an edge-using segment. Similarly, if the absolute value of an out-angle is $\pi/2$ then the segment after a is an edge-using segment. In the case when Π is a geodesic path, angles φ and ψ are related as follows:

LEMMA 2.1. *(Snell's law) Let a be a bending point of a geodesic path Π that lies inside an edge e of $\mathcal{P}$. Let the segment preceding a in Π be s' and the one after a be s''. Let the costs of traveling along s' and s'' be w' and w'' respectively. Then*

$$w''\sin\psi=w'\sin\varphi, \tag{2.1}$$

where φ and ψ are the in-angle and out-angle at a.

In the next lemma we establish an estimate on the difference between out-angles of two given geodesic paths in terms of the difference between their in-angles. The result is somewhat counter intuitive as one may have expected that $|\varphi_1-\varphi|\le\epsilon$ implies $|\psi_1-\psi|\le\epsilon * constant$, for all angle pairs satisfying Snell's law and all ϵ, where the constant only depends on face weights.

LEMMA 2.2. *Let φ, ψ and φ_1, ψ_1 be two pairs of angles each satisfying (2.1). Assume that $w'>w$". Then*

$$|\psi_1-\psi|\le\frac{\pi w'}{2w''}\kappa(\varphi,\varphi_1)|\varphi_1-\varphi|, \tag{2.2}$$

where $\kappa(\varphi,\varphi_1)=\cos\frac{\varphi_1+\varphi}{2}/\cos\frac{\psi_1+\psi}{2}$.

2.2 Discretization:

DEFINITION 2.1. *Given a point $x\in\mathcal{P}$ the distance $d(x)$ of x to the set of edges of $\mathcal{P}$ is defined as the Euclidean distance $d(x)$ on $\mathcal{P}$ from x to $E\setminus E(x)$, where E is the set of the edges of $\mathcal{P}$ and $E(x)$ are the edges incident to x.*

For each vertex v of $\mathcal{P}$ we define a weighted radius

$$r(v)=\frac{w_{\min}(v)}{5w_{\max}(v)}d(v), \tag{2.3}$$

where $w_{\max}(v)$ and $w_{\min}(v)$ are the maximum and minimum weights of the faces incident to v. By using the weighted radius $r(v)$ for each face incident to v we define a "small" isosceles triangle with two sides of length $\varepsilon r(v)$ incident to v. All these small triangles around v form a star shaped polygon $S(v)$ which we call a *vertex vicinity* of v.

Next, we describe the placement of Steiner points on edges, e, of $\mathcal{P}$. Assume that $e=(v',v'')$ and let $d(e)=\sup_{x\in e}d(x)$. Let M be a point on e, so that $d(M)=d(e)$. Such a point M can be easily computed in time proportional to the size of the two faces incident to e. The Steiner points on e are defined as follows. The point M is a Steiner point. On the edge-segment (v',M) we define points $p'_1,\ldots,p'_{k'}$ so that $|v'p'_1|=\varepsilon r(v')$, and $|p'_{i-1}p'_i|=\varepsilon d(p'_{i-1})$, for $i=2,\ldots k'$. Note that the first Steiner point p'_1 coincides with the vertex of the vertex vicinity $S(v')$ that lies on e. The Steiner points $p''_1,\ldots,p''_{k''}$ on the edge-segment (v'',M) are defined in the same manner. To simplify notation, we denote Steiner points on e following their order from v' to v'' by $p_1,\ldots,p_k$, where $k=k'+k''+1$ and $M=p_{k'+1}$. Using the above procedure we insert Steiner points on all edges of $\mathcal{P}$. In the next lemma we estimate (quite precisely) the number of Steiner points inserted on an edge of $\mathcal{P}$ and consequently the total number of Steiner points on $\mathcal{P}$. We denote by $E(\mathcal{P})$ the set of edges of $\mathcal{P}$.

LEMMA 2.3. *(a) The number of Steiner points inserted on an edge $e=(v',v'')$ does not exceed $3+C(e)\frac{1}{\varepsilon}\log_2\frac{2}{\varepsilon}$, where the constant $C(e)<4(|e|/d(e))\log_2(|e|/\sqrt{r(v')r(v'')})$.*

(b) The total number of the Steiner points inserted on $\mathcal{P}$ is bounded by $\tilde{C}(\mathcal{P})\frac{m}{\varepsilon}\log_2\frac{2}{\varepsilon}$, where $\tilde{C}(\mathcal{P})-3$ does not exceed the average of the constants $C(e)$ on $E(\mathcal{P})$, i.e., $\tilde{C}(\mathcal{P})\le 3+\frac{1}{m}\sum_{e=(v',v'')\in E(\mathcal{P})}C(e)$.

The set of Steiner points defines a set of *Steiner intervals.* The essential property of the Steiner point set on $\mathcal{P}$ is that any face-crossing segment s having an endpoint in a Steiner interval (p_i,p_{i+1}), $1\le i\le k-1$, satisfies the inequality

$$|p_ip_{i+1}|\le\varepsilon|s|. \tag{2.4}$$

The above property provides an upper bound on the maximum angle θ at which a Steiner interval (p_i,p_{i+1}) is seen from a point on the boundary of the quadrilateral formed by the two triangles neighboring that interval.

LEMMA 2.4. *Let (p_i, p_{i+1}) be a Steiner interval on an edge e and x be a point on the boundary of the union of the two faces neighboring e. Then*

$$\angle p_i x p_{i+1} \leq \frac{\pi}{2}\varepsilon. \tag{2.5}$$

Next we show that the density of the Steiner points is sufficient to approximate face-crossing segments with segments joining pairs of Steiner points with ε accuracy. More precisely, for a face-crossing segment (a, b) that does not intersect vertex vicinities we show that:

LEMMA 2.5. *Let (a, b) be a face-crossing segment that does not belong to a vertex vicinity.*
a) *If (a, b) joins a pair of Steiner intervals say (p_i, p_{i+1}) and (q_j, q_{j+1}) then*

$$\max\left[\min(|p_i q_j|, |p_i q_{j+1}|), \min(|p_{i+1} q_j|, |p_{i+1} q_{j+1}|)\right] \leq (1+\varepsilon)|ab|. \tag{2.6}$$

b) *If (a, b) joins a segment (v, p_1) between a vertex v and a Steiner point p_1 with a Steiner interval (q_j, q_{j+1}) then*

$$\max\left[(|p_1 q_j|, |p_1 q_{j+1}|)\right] \leq (1+\varepsilon)|ab| + \varepsilon r(v) \tag{2.7}$$

c) *If (a, b) joins two segments (v, p_1) and (v', q_1), where v and v′ are vertices and p_1, q_1 are their neighbour Steiner points then*

$$|p_1 q_1| \leq |ab| + \varepsilon(r(v) + r(v')) \tag{2.8}$$

2.3 Discrete paths: In Section 2.2 we have described a particular method for placing Steiner points on edges of $\mathcal{P}$. This suggests the definition of a graph $G_\varepsilon = (V(G_\varepsilon), E(G_\varepsilon))$ whose vertex set $V(G_\varepsilon)$ includes all Steiner points plus the vertices of $\mathcal{P}$. The set of edges of $E(G_\varepsilon)$ consists of all face-crossing segments joining pairs of Steiner points plus edge-using segments joining either a pair of neighboring Steiner points or a vertex of $\mathcal{P}$ with its neighboring Steiner point. Now any path in G_ε is referred to as a *discrete path* on $\mathcal{P}$. Note that discrete paths are linear as well.

An *ε-short* path for a path $\tilde{\Pi}$ on $\mathcal{P}$ is a discrete path whose cost is at most $(1+\varepsilon)\|\tilde{\Pi}\|$. In this section we first show the existence of a 3ε-short discrete path for any linear path on $\mathcal{P}$. Then, by using a spanner technique, we define a subgraph G^*_ε of G_ε that has a smaller set of edges but still contains 7ε-short discrete path for any linear path. The graph G^*_ε will be used in the next section where we present a pruned Dijkstra algorithm for approximating geodesic paths on $\mathcal{P}$.

Next we introduce some notions about the structure of linear paths as necessary for our considerations. Consider a linear path $\tilde{\Pi}(v_0, v)$ joining two different vertices of $\mathcal{P}$. This path crosses vertex vicinities $S(v_0)$ and $S(v)$ and possibly a number of other vertex vicinities. The path $\tilde{\Pi}$ can be partitioned into path portions $\tilde{\Pi}(S(v_i))$, lying fully inside *vertex vicinities*, $S(v_i)$, and those, called *between-vertex-vicinities* $\tilde{\Pi}(S(v_i), S(v_{i+1}))$, connecting vertex vicinities $S(v_i)$ and $S(v_i + 1)$. [2] Assume now that the path $\tilde{\Pi}(v_0, v)$ consists of $k+2$ vertex-vicinity and $k+1$ between-vertex-vicinity portions.

Next, we introduce the notion of a discrete path that *neighbours* a given linear path $\tilde{\Pi}(v_0, v)$. We describe first a discrete path that neighbours a between-vertex-vicinities portion. Consider a between-vertex-vicinity portion $\tilde{\Pi}(S(v_i), S(v_{i+1})$ for some $0 \leq i \leq k$. This portion consists of a sequence of face-crossing and edge-using segments that do not intersect vertex vicinities. Each of these segments joins two Steiner intervals and thereafter the portion $\tilde{\Pi}(S(v_i), S(v_{i+1})$ defines a sequence of Steiner intervals. A discrete path $\Pi(S(v_i), S(v_{i+1})$ neighbours the between-vertex-vicinities linear path $\tilde{\Pi}(S(v_i), S(v_{i+1})$ if it joins a Steiner point on the boundary of the vertex vicinity $S(v_i)$ with a Steiner point on the boundary of $S(v_{i+1})$ and crosses the same sequence of Steiner intervals. Informally, when the linear path crosses a Steiner interval its discrete neighbour snaps to one of the Steiner points forming that interval. In general, we use the following definition.

DEFINITION 2.2. *A discrete path $\Pi(v_0, v)$ is a neighbour of the linear path $\tilde{\Pi}(v_0, v)$ given as above, if its between-vertex-vicinities portions neighbour corresponding between-vertex-vicinities portions of $\tilde{\Pi}$ and its vertex-vicinity portions are two edge paths joining between-vertex-vicinities portions through the corresponding vertex.*

Now we state and prove (see the Appendix also for a formal definition) the central theorem of this section.

THEOREM 2.1. *Let $\tilde{\Pi}(v_0, v)$ be a linear path joining two different vertices on $\mathcal{P}$. There exists a discrete 3ε-short path $\Pi(v_0, v)$ that neighbours $\tilde{\Pi}$.*

Theorem 2.1 provides a method for approximating geodesic paths on $\mathcal{P}$. We may consider the graph $G_\varepsilon = (V(G_\varepsilon), E(G_\varepsilon))$ and assign weights on its edges equal to the cost of traveling along them. Then by Theorem 2.1 the cost of a shortest path in G_ε between any two of its nodes does not exceed $(1 + 3\varepsilon)$ times the cost of the geodesic path on $\mathcal{P}$. Then Dijkstra's

[2] Note that according to our definition a vertex-vicinity portion $\tilde{\Pi}(S(v))$ can be empty although the path crosses $S(v)$ as well it can contain segments outside $S(v)$ that lie between the first and the last intersection point of $\tilde{\Pi}$ and $S(v)$.

algorithm can be used for finding shortest paths in G_ε in time $O(|E(G_\varepsilon)| + |V(G_\varepsilon)| \log |V(G_\varepsilon)|)$. Therefore, we may approximate geodesic paths on $\mathcal{P}$ in time $O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon}(\frac{1}{\varepsilon} \log \frac{1}{\varepsilon} + \log(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon})))$.

Corollary 2.1. *For $\varepsilon = \frac{1}{3}$ we obtain an $O(n \log n)$ algorithm that finds a path of cost at most twice that of the shortest.*

A weighted shortest path can have $O(\Omega(n^2))$ segments. This does however not contradict our time bounds as in such a case many segments will cross the same Steiner interval and are thus replaced by a single approximating segment. For the remainder we will assume that $\varepsilon < 1$. Note however that this approach does not use any properties of geodesic paths on $\mathcal{P}$. It turns out that using the additional information about the geometry of the geodesic paths (Lemma 2.1) we may prove a useful characterization of the paths in G_ε that neighbour geodesic paths on $\mathcal{P}$ and thereafter direct our search. This will result in an $O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}} + \log n))$ time approximation algorithm.

Definition 2.3. *A discrete path $\Pi(v_0, v)$ is a discrete geodesic path if it neighbours a geodesic path joining v_0 and v.*

Theorem 2.1 directly implies the following corollary.

Corollary 2.2. *For any geodesic path $\tilde{\Pi}(v_0, v)$ there exist a discrete geodesic path $\Pi(v_0, v)$ such that $\|\Pi\| \leq (1 + 3\varepsilon)\|\tilde{\Pi}\|$.*

Corollary 2.2 suggests that in order to approximate geodesic paths on $\mathcal{P}$ it suffices to search in the class of discrete geodesic paths. So we need an efficient procedure to compute and search that class. First we show that discrete geodesic paths satisfy a constraint similar to those established for geodesic paths in Lemma 2.1. More precisely given a discrete geodesic path Π and a segment (a, b) of Π, we can prove that there is a cone $C(a, b)$ with apex b and angle depending on ε and weights around b, that contains the next bending point c of Π. Note that in the case of geodesic paths any of its segments completely determines the direction of its next segment, i.e., the angle of the corresponding cone is zero.

Let $\Pi(v_0, v)$ be a discrete geodesic path. By the definition there is a geodesic path neighboring Π. Let a, b, and c be three consecutive bending points of Π, where b does not neighbour a vertex of $\mathcal{P}$. Here a and c are Steiner points (not vertices of $\mathcal{P}$). According to Lemma 2.1, the segment (a, b) defines a unique direction at which a geodesic path comprising (a, b) must continue after b. Except for the case where that direction is along the edge containing b, a unique face-crossing segment (b, c_0) is defined so that the in-angle φ and the out-angle ψ at of the sub-path $\{a, b, c_0\}$ satisfy Lemma 2.1. The next lemma establishes an estimate on the size of the angle $\angle cbc_0$.

Lemma 2.6. *Let $\varepsilon < 1$ and a, b and c be three consecutive bending points on a discrete geodesic path Π, where the Steiner point b does not neighbour a vertex of $\mathcal{P}$. Let c_0 be a point on the geodesic direction determined by the segment (a, b) (on the ray starting at b and with an out-angle ψ that satisfies Lemma 2.1). Then*

$$\angle cbc_0 \leq \pi(1 + \sqrt{\frac{\pi w'}{2w''}})\sqrt{\varepsilon}, \tag{2.9}$$

w', w'' are the weights of the faces incident to b.

The result of Lemma 2.6 can be summarized in the following way. If Π is a discrete geodesic path and (a, b) is a segment of Π, so that b does not neighbour a vertex of $\mathcal{P}$ then the cone defined using (2.9) contains the next segment(b, c) of Π. Hereafter, we call the set of edges (b, c) that satisfy (2.9) with respect to (a, b) *geodesic cone* and denote it by $GCone(a, b)$. Note that if b neighbours or is a vertex $\mathcal{P}$ then the lemma provides no information and we assume, that $GCone(a, b)$ consists of all edges incident to b except (a, b).

Let a_0 be any fixed node of G_ε. Based on Lemma 2.6 we define a class of paths $\mathcal{C}(a_0)$ consisting of all discrete paths that originate from a_0 and satisfy Lemma 2.6. Clearly, $\mathcal{C}(a_0)$ contains discrete geodesic paths originating from a_0. An inductive description of $\mathcal{C}(a_0)$ is as follows:

Definition 2.4. *The class of paths $\mathcal{C}(a_0)$ is defined by:*

1. *Edges of G_ε incident to a_0 belong to $\mathcal{C}(a_0)$.*
2. *A discrete path $\Pi' = \{a_0, a_1, \ldots, a_{k-1}, a_k\}$, $k > 1$ belongs to $\mathcal{C}(a_0)$ iff $\Pi = \{a_0, a_1, \ldots, a_{k-1}\} \in \mathcal{C}(a_0)$ and $(a_{k-1}, a_k) \in GCone(a_{k-2}, a_{k-1})$.*

2.4 A Pruned Dijkstra Algorithm: We have shown, that geodesic paths originating from a fixed node a_0 of G_ε to all other nodes can be approximated by finding single source shortest paths in $\mathcal{C}(a_0)$. We may do that by employing a *pruned Dijkstra* algorithm whose search is restricted to the class $\mathcal{C}(a_0)$. Note that the inductive definition of $\mathcal{C}(a_0)$ perfectly meets that purpose. However, we cannot prove the claimed efficiency bound since the number of the edges in cones $Cone(a, b)$ still can be large. To obtain an efficient algorithm we define and employ a class of paths $\mathcal{C}^*(a_0)$ that contains an ε-approximation path for each path in

$\mathcal{C}(a_0)$. Additionally the class $\mathcal{C}^*(a_0)$ will be defined so that the pruned Dijkstra algorithm will perform fewer updates per step while searching it. The definition of $\mathcal{C}^*(a_0)$ resembles the Definition 2.4, but here the sets of edges, $Cone^*(\Pi)$, that can continue a given path $\Pi \in \mathcal{C}^*(a_0)$ are subsets of the cones from Definition 2.4 of size $O(\varepsilon^{-\frac{1}{2}})$. We specify sets $Cone^*(\Pi)$ in detail after the definition.

DEFINITION 2.5. *The class $\mathcal{C}^*(a_0)$ consists of:*

1. *Edges of G_ε incident to a_0 are one edge paths in $\mathcal{C}^*(a_0)$.*
2. *A discrete path $\Pi' = \{a_0, a_1, \ldots, a_{k-1}, a_k\}$, $k > 1$ belongs to $\mathcal{C}^*(a_0)$ iff*

$$\Pi = \{a_0, a_1, \ldots, a_{k-1}\} \in \mathcal{C}^*(a_0) \quad \text{and} \quad (a_{k-1}, a_k) \in Cone^*(\Pi). \tag{2.10}$$

Next, given a path $\Pi = \{a_0, a_1, \ldots, a_{k-1},\}$, $k > 1$ in $\mathcal{C}^*(a_0)$ we define the set $Cone^*(\Pi) = Cone^*(\Pi, a_{k-1})$. Intuitively, it consists of an ε-spanner of some of geodesic cones $Cone(a_{k-i}, a_{k-1})$, $i > 1$. Let us introduce first the notion of an ε-spanner of a cone of edges. A cone $Cone(b)$ with apex b is a set of all edges incident to b that belong to the intersection of one of the faces around b. Otherwise, $Cone(b)$ is a sequence of consecutive edges around b lying in one of the faces that neighbour b. The angle between the first and the last edge in the sequence $Cone(b)$ is called the angle of $Cone(b)$. Note that by our definition the angle of a cone can not exceed π.

An ε-spanner $Cone_\varepsilon(b)$ of $Cone(b)$ with an angle $\theta \in (0, \pi)$ is defined as a subset of at most $\lceil 2\theta/\pi\varepsilon \rceil$ edges of $Cone(b)$ as follows. The $Cone(b)$ is partitioned into $j \le \lceil 2\theta/\pi\varepsilon \rceil$ sub-cones $Cone_1(b), \ldots, Cone_j(b)$, so that each of these cones contains at least one edge and their angles do not exceed $\pi\varepsilon/2$. Such a partition is always possible as a result of Lemma 2.4. Then $Cone_\varepsilon(b)$ is defined as the set of the shortest length edges $s_1, \ldots, s_j$, in the sub-cones $Cone_1(b), \ldots, Cone_j(b)$. The lemma below states an important property of ε-spanners.

LEMMA 2.7. *Let $Cone(b)$ and $Cone_\varepsilon(b)$ be a cone of edges and its ε-spanner. For any edge (b, c) in $Cone(b)$ there is an edge (b, c') in $Cone_\varepsilon(b)$ so that*

$$|bc'| \le |bc| \le |bc'| + |cc'| \le (1 + \frac{\pi\varepsilon}{2})|bc|. \tag{2.11}$$

The following cases arise for the last node and edge of the path $\Pi = \{a_0, a_1, \ldots, a_{k-1}\}$. We only discuss the most general case (3.2) and leave the special case discussion for the Appendix.

***Case 1*:** The node a_{k-1} is a vertex of $\mathcal{P}$.
***Case 2*:** The node a_{k-1} neighbours a vertex of $\mathcal{P}$.
***Case 3*:** The node a_{k-1} neither neighbours nor is a vertex of $\mathcal{P}$.
We consider the two possible sub-cases.
Case 3.1: The edge (a_{k-2}, a_{k-1}) is an edge-using segment.

Case 3.2: The edge (a_{k-2}, a_{k-1}) is a face-crossing segment. In this case the set $Cone^*(\Pi, a_{k-1})$ consists of the two edges joining a_{k-1} with its neighboring Steiner points plus the ε-spanner of the geodesic cone $GCone(a_{k-2}, a_{k-1})$. The angle of the geodesic cone was estimated in Lemma 2.6 and according to the construction of ε-spanner and (2.9) the size of $Cone^*(\Pi, a_{k-1})$ is at most

$$|Cone^*(\Pi, a_{k-1})| \le 3 + \pi^2(1 + \sqrt{\frac{\pi w'}{2w''}})\varepsilon^{-\frac{1}{2}}, \tag{2.12}$$

where w' is the weight of the face containing (a_{k-2}, a_{k-1}) and w'' is the weight of the other face incident to a_{k-1}.
In the next lemma we establish that paths in the class $\mathcal{C}(a_0)$ are approximated by paths in $\mathcal{C}^*(a_0)$.

LEMMA 2.8. *Let Π be a discrete geodesic path starting at a_0 and consisting of k edges $\Pi = \{a_0, a_1, \ldots, a_k\}$. The class $\mathcal{C}^*(a_0)$ contains a path Π^* of the form $\Pi^* = \{a_0, a_1, \ldots, a_2, \ldots, a_{k-1}, \ldots, a_k\}$, where each of the subpaths $\Pi^*(a_{i-1}, a_i)$ for $i = 1, \ldots, k$ consists either of a single edge (a_{i-1}, a_i) or of a single face-crossing edge (a_{i-1}, a'_i) plus a sequence of edges joining consecutive Steiner points between a'_i and a_i along an edge of $\mathcal{P}$. The cost of the path Π^* satisfies*

$$\|\Pi^*\| \le (1 + \pi\varepsilon/2)\|\Pi\|. \tag{2.13}$$

Next we present an efficient algorithm for finding shortest paths in the class $\mathcal{C}^*(a_0)$ from a_0 to all other nodes of G_ε. Actually, this is Dijkstra's single source shortest paths algorithm restricted to search in the class $\mathcal{C}^*(a_0)$ only. The algorithm takes as input the surface $\mathcal{P}$, the set of Steiner points V_ε as defined in Section 2.2 and a fixed Steiner point a_0. It outputs a tree $SPT(\mathcal{C}^*(a_0))$ routed at a_0 and consisting of single source shortest paths in the class $\mathcal{C}^*(a_0)$. The only difference between the algorithm and the classic Dijkstra's algorithm is that this one does not update all possible continuations of the latest output shortest path but only the nodes forming paths in $\mathcal{C}^*(a_0)$ using edges $(a, b) \in Cone^*(\Pi(a), a)$. As the classical Dijkstra's algorithm, our modified algorithm, called PrunedDjikstra, employs a priority queue Q.

We may implement Q (see, [16]) so that the amortized time for each update in Step 2.2 is a constant and therefore it is proportional to the size of the corresponding set $Cone^*(\Pi(a), a)$ if a was extracted from the queue. According to the definition of the sets $Cone^*(\Pi(a), a)$ their sizes are:

1. $|Cone^*(\Pi(a), a)| = O(\varepsilon^{-\frac{1}{2}})$ if a is a node that does not neighbour nor is a vertex of $\mathcal{P}$.
2. $|Cone^*(\Pi(a), a)| = O(\varepsilon^{-1})$ if a neighbours a vertex of $\mathcal{P}$.
3. $|Cone^*(\Pi(a), a)| = \deg(a)$ if a is a vertex of $\mathcal{P}$, where $\deg(a)$ denotes the number of edges incident to a in $\mathcal{P}$.

Therefore the total time for the implementation is proportional to the $\sum_{a\in V_\varepsilon} |Cone^*(\Pi(a), a)|$ which equals $O(\frac{n}{\varepsilon^{\frac{3}{2}}} \log \frac{1}{\varepsilon})$.

LEMMA 2.9. *Algorithm PrunedDjikstra outputs a rooted spanning tree $SPT(\mathcal{C}^*(a_0))$ of G_ε and for each $a \in V_\varepsilon$, the cost of the shortest path from a_0 to a in the class $\mathcal{C}^*(a_0)$. The algorithm runs in $O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}} + \log n))$ time.*

In summary, given a polyhedral surface $\mathcal{P}$ as defined in Section 2.1 and an approximation parameter $\varepsilon > 0$ we discretize $\mathcal{P}$ by adding a set of Steiner points as described in Section 2.2. Recall that we denoted the set of Steiner points plus the vertices of $\mathcal{P}$ by V_ε. Then we consider a fixed source point $a_0 \in V_\varepsilon$ and let $Cost(a; \mathcal{P})$ denote the cost of the shortest path from a_0 to $a \in V_\varepsilon$ on $\mathcal{P}$. To find approximate shortest paths we simply run Algorithm PrunedDjikstra above and report the paths $\Pi^*(a_0, a)$ from the spanning tree $SPT(\mathcal{C}^*(a_0))$ computed by the algorithm. According to Lemma 2.9, Lemma 2.8 and Theorem 2.1 we have

$$\begin{aligned}
&\|\Pi^*(a_0, a)\| = \min\{\|\Pi\| : \Pi \in \mathcal{C}^*(a_0)\} \\
&\le (1 + \pi\varepsilon/2) \min\{\|\Pi\| : \Pi \in \mathcal{C}(a_0)\} \\
&\le (1 + 3\varepsilon)(1 + \pi\varepsilon/2) Cost(a; \mathcal{P}) \\
&\le (1 + 15\varepsilon) Cost(a; \mathcal{P})
\end{aligned}$$

Thus we have established the following theorem:

THEOREM 2.2. *Let $\mathcal{P}$ be a weighted polyhedral surface and $0 < \varepsilon \le 1$. The single source shortest paths on $\mathcal{P}$ can be approximated with accuracy factor $(1 + 15\varepsilon)$ in $O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}} + \log n))$ time.*

3 WRP-3D Problem

3.1 Introduction: This section presents an approximation scheme on a given triangulation (tetrahedralization) $\mathcal{D}$ in three dimensional Euclidean space $\Re^3$ consisting of edges, faces and tetrahedra. The domain is discretized by placing a set of Steiner points on the faces and edges of the tetrahedra and inserting a set of segments across tetrahedra that interconnects pairs of Steiner points. The placement and the number of Steiner points depends upon the desired approximation factor ε as well as the geometry and weights of the given configuration. Following the approach taken in solving SPPS problem, to solve WRP-3D we need to address a) the placement of Steiner points, b) the discretization of Snell's law in three dimensional space, and c) the approximation quality of paths computed by Pruned Dijkstra's Algorithm. Due to the lack of space, we only sketch the main ideas here and refer the reader to the full version.

3.2 Preliminaries: Let $\mathcal{D}$ be a connected triangulation of space in three dimensions, with m edges, f faces (triangles) and n tetrahedra $t_1, \ldots, t_n$. Positive weights $w_1, \ldots, w_n$ are associated with tetrahedra representing the costs of traveling inside them. The cost of traveling along a face or an edge of a tetrahedron is the minimum of the weights of the neighboring tetrahedra. The cost of a path π on $\mathcal{D}$ is defined by $\|\pi\| = \sum_{i=1}^n w_i|\pi_i|$, where $|\pi_i|$ denotes Euclidean length of the intersection $\pi_i \cap t_i$. Given two vertices u and v in $\mathcal{D}$ the geodesic path $\pi(u, v)$ between them is the path with minimum cost joining them and remaining in $\mathcal{D}$. As for 2-D, it is well known that geodesic paths are simple. The segments in a geodesic path can be of the following types: 1) *cell-crossing* segments which cross a tetrahedra joining two points on its boundary that lie on a different faces, 2) *face-using* segments which lie along a face, and 3) *edge-using* segments which lie along an edge. Consecutive segments of a geodesic path obey Snell's law of refraction in which the path bends at the faces and edges of $\mathcal{D}$. Steiner points in $\mathcal{D}$ will be defined with the aid of the following:

DEFINITION 3.1. *For a point $x \in \mathcal{D}$, define $d(x)$ to be the Euclidean distance from x to the boundary of the union of the tetrahedra incident to x.*

For example, if x lies on a face shared by two tetrahedra, then $\delta(x)$ is the distance from x to the boundary of the union of these two tetrahedra. For a vertex v of $\mathcal{D}$ we define a radius $r(v) = \frac{w_{\min}(v)}{c w_{\max}(v)} d(v)$, where $w_{\min}(v)$ ($w_{\max}(v)$) is the minimum (maximum) weight around v and $c \ge 1$ is a constant. Similarly, for a point x on an edge e (not a vertex) of $\mathcal{D}$ we define a radius $r(x) = \frac{w_{\min}(e)}{c w_{\max}(e)} d(x)$, where $w_{\min}(e)$ ($w_{\max}(e)$) is the minimum (maximum) weights around e. Also we define $r(e)$ to be the maximum of $r(x)$ for $x \in e$, i.e. $r(e) = \max_{x \in e} r(x)$.

DEFINITION 3.2. *A vertex-vicinity $\mathcal{V}(v)$ is the ball, $B(v, \varepsilon r(v))$, centered at v of radius $\varepsilon r(v)$. For an edge $e = (v', v'')$ we define an edge-vicinity $\mathcal{V}(e) = \mathcal{V}(v') \cup \mathcal{V}(v'') \cup \mathcal{S}(e)$, where the* spindle *$\mathcal{S}(e)$ of e is defined by $\mathcal{S}(e) = \cup_{x \in e} B(x, \varepsilon r(x))$.*

3.3 Placement of Steiner Points: The placement of Steiner points is more complex compared to the placement in the SPPS problem since, a shortest path may bend in the interior of a face. We therefore need to take care of edge vicinities in addition to the vertex vicinities. An infinitesimal segment may penetrate through these vicinities and any placement of a finite set of Steiner points and edges cannot provide good approximations to these segments. This is addressed by providing an amortization argument for the cost of the part of the path inside these vicinities with respect to the total cost of the path.

Let f be a face with vertices A, B and C and let M be the point in f with maximum distance $d(M)$, i.e., $d(M) = \max_{x \in f} d(x)$. The point M is defined to be a Steiner point. We describe the placement of Steiner points inside the triangle ABM. Let MH be the height of the triangle ABM. Define an infinite sequence of points $P_0, P_1, \dots$ on MH as follows: $P_0 = M$ and for $i = 1, \dots$, $|P_{i-1}P_i| = \varepsilon d(P_i)$. Consider the corresponding sequence of segments A_iB_i that are parallel to AB and contain P_i for $i = 1, \dots$ On each of the segments A_iB_i define a set of $k_i + 1$ equi-distantly placed points $P_{i,j}$, $j = 0, \dots, k_i$ with

$$k_i = \lceil |A_i, B_i| / (\varepsilon d(P_{i+1})) \rceil. \tag{3.14}$$

The formal description of these points is given by $P_{i,0} = A_i$, $P_{i,k_i} = B_i$ and for $j = 1, \dots, k_i$, $|P_{i,j-1}P_{i,j}| = |A_iB_i|/k_i$.

DEFINITION 3.3. *The set of Steiner points in the triangle ABM consists of all points $P_{i,j}$ that are outside the edge-vicinity $\mathcal{V}(AB)$.*

The total number of Steiner points placed in $\mathcal{D}$ is given by the following lemma whose proof can be found in the Appendix.

LEMMA 3.1. *The number of Steiner points placed in a face f of $\mathcal{D}$ does not exceed $C_f(\mathcal{D}) \frac{1}{\varepsilon^3} \ln \frac{1}{\varepsilon}$, where the constant $C_f(\mathcal{D})$ depends on features of $\mathcal{D}$ around f. The total number of Steiner points is bounded by $C(\mathcal{D}) \frac{n}{\varepsilon^3} \ln \frac{1}{\varepsilon}$, where n is the number of faces of $\mathcal{D}$ and the constant $C(\mathcal{D})$ is the arithmetic average of the constants $C_f(\mathcal{P})$ for the faces of $\mathcal{D}$.*

3.4 Forming the approximation graph: Consider a segment ab of a geodesic path lying inside a trapezoid t_i, such that a and b belong to two different faces f_1 and f_2 of t_i, respectively. Assume that ab does not intersect any vertex or edge vicinities. It can be shown that there exist Steiner points $a' \in f_1$ and $b' \in f_2$ so that weighted length of the path from a to b through a' and b' is at most $1 + \varepsilon$ times the cost of the direct path from a to b. This enables us to provide ε approximations to the part of the geodesic path that does not lie inside vertex or edge vicinities. The part of the geodesic path that lies inside vertex vicinities can be handled in a similar manner as in the SPPS problem, i.e., amortizing its cost with respect to the cost of the path that is outside the vertex vicinities. Now concentrate on the part of the path (say $\Pi(e)$) that lies inside an edge vicinity $\mathcal{V}(e)$. There are two cases depending on the cost of $\Pi(e)$ with respect to the parameter $\varepsilon r(x)$ (Definition 3.2). If the two costs are comparable then use the same amortization argument as in the case of vertex vicinities. If the cost of $\Pi(e)$ is more than $\varepsilon r(x)$ then there exist a pair of Steiner points a' and b' on the boundary of $\mathcal{V}(e)$, such that the cost of the path between these Steiner points is at most $1 + \varepsilon$ times the cost of $\Pi(e)$.

The above discussion suggests that if we interconnect all Steiner points within a trapezoid, for every trapezoid in $\mathcal{D}$, and interconnect all Steiner points on the boundary of edge vicinities, then we can obtain a graph G, where vertices are Steiner points, and the edges are the interconnections. For any linear path $\tilde{\Pi}$ joining a pair of vertices of $\mathcal{D}$, it follows that there exists a discrete path Π between the corresponding vertices in G whose cost is at most $1 + \varepsilon$ times the cost $\tilde{\Pi}$. Although the approximation quality of G is the desired one, the number of edges, and hence the computational complexity of finding a shortest path in G, is very high. Next we show that using properties of Snell's law, the search can be restricted to a selected set of edges when a vertex is "explored" (after extract-min) in an execution of Dijkstra's algorithm.

3.5 Pruned Dijkstra's algorithm: Let a be a bending point on a geodesic path Π lying in the interior of a face f. The in-angle φ is defined to be the acute clockwise angle between the normal to f at a and the segment s' preceding a on Π. Similarly, the out-angle ψ is the acute clockwise angle between the normal and the segment s'' succeeding a on Π. These angles are related by $w'' \sin \psi = w' \sin \varphi$, where the cost of traveling along s' and s'' is w' and w'', respectively. Moreover the segments s', s'', and the normal to f at a (denoted by $\mathcal{N}_{fb}$) are in the same plane. This can be expressed by taking a reference direction in f, and the angles $\eta_{in} = \eta_{out}$, where η_{in} (η_{out}) is the angle between the projection of s' (s'') on f and the reference direction.

Next we find an estimate on the difference between out-angles of two given geodesic paths in terms of the difference between their in-angles.

LEMMA 3.2. *Let* $(\varphi, \psi, \eta_{in}, \eta_{out})$ *and* $(\varphi_1, \psi_1, \eta_{1in}, \eta_{1out})$, *be two pairs of angles satisfying the above equations. Then*

$$(a) \quad |\psi_1 - \psi| \leq \frac{\pi w'}{2w''} \kappa(\varphi, \varphi_1)|\varphi_1 - \varphi| \tag{3.15}$$

$$(b) \quad |\eta_{1in} - \eta_{in}| = |\eta_{1out} - \eta_{out}| \tag{3.16}$$

where $\kappa(\varphi, \varphi_1) = \cos\frac{\varphi_1+\varphi}{2} / \cos\frac{\psi_1+\psi}{2}$.

The next lemma estimates the dispersion of a discrete path with respect to the geodesic path (equivalent to Lemma 2.6).

LEMMA 3.3. *Let* $\varepsilon < 1$ *and* a, b *and* c *be three consecutive bending points on a discrete path* Π, *where Steiner point* b *lying in the face* f *does not neighbour a vertex or edge vicinity of* $\mathcal{D}$. *Let* c_0 *be a point on the geodesic direction determined by the segment* (a, b) *(on the ray starting at* b *and with an out-angle* ψ, η_{out} *satisfying Snell's law). Then, (a) the angle between the two planes determined by* $\mathcal{N}_{fb}$ *and* bc_0 *and* bc *is at most* ε. *(b) the angle between the projection of* bc, *onto the plane defined by the* bc_0 *and* $\mathcal{N}_{fb}$, *and* bc_0 *is at most* $c'\sqrt{\varepsilon}$, *where* c' *is a constant.*

The above lemma establishes that during an execution of Dijkstra's algorithm, it is not necessary to look at all edges in the approximation graph G, but rather look at those edges which are given by the specified angle constraint. Following along the lines of Section 2.4, it can be shown that for all Steiner points which are not neighboring vertex or edge vicinities, we have to explore only $O(1/\sqrt{\varepsilon})$ edges. We summarize the result in the following theorem:

THEOREM 3.1. *Let* $\mathcal{D}$ *be a weighted triangulation of 3-dimensional space and* $0 < \varepsilon < 1$ *consisting of* n *vertices. The single source shortest paths on* $\mathcal{D}$ *can be approximated to within an accuracy of* $(1 + c\varepsilon)$ *in* $O(\frac{n}{\varepsilon^3} \log\frac{1}{\varepsilon}(\frac{1}{\sqrt{\varepsilon}} + \log n))$ *time, where* $c \geq 1$ *is a constant.*

In case all the weights are the same, the complexity reduces to $O(\frac{n}{\varepsilon^3} \log\frac{1}{\varepsilon} \log n)$ time. This can used to solve the shortest path problem amidst obstacles in 3 dimensional Euclidean space (ESP-3D), provided that we are given a convex partition (or tetrahedralization) of the free space. Convex partition can be achieved by employing any one of the algorithms of [19, 4, 14].

References

[1] K.R. Varadarajan and P.K. Agarwal, "Approximating Shortest Paths on a Polyhedron",*38FOCS*,1997.

[2] P.K. Agarwal, S. Har-Peled, M. Sharir, and K.R. Varadarajan, "Approximating Shortest Paths on a Convex Polytope in Three Dimensions",*J. ACM*, 1998.

[3] L. Aleksandrov, M.Lanthier, A. Maheshwari, J.-R. Sack, "An ε-Approximation Algorithm for Weighted Shortest Path Queries on Polyhedral Surfaces" *14th Euro. Workshop on Comp. Geom.*, Spain, 19-21, 1998.

[4] C. Bajaj and T. Dey, "Convex decompositions of polyhedra and robustness", SIAM Jl. Comput. 21, 1992, pp. 339-364.

[5] A. Baltsan and M. Sharir, "On the Shortest Paths Between Two Convex Polyhedra", *J. ACM*, **35**, 1988, pp. 267-287.

[6] J. Canny and J. H. Reif, "New Lower Bound Techniques for Robot Motion Planning Problems",*28th IEEE Symp. on Foundations of Computer Science*, 1987, pp. 49-60.

[7] J. Chen and Y. Han, "Shortest Paths on a Polyhedron", *6th SoACM-CG*, 1990, pp. 360-369.

[8] K.L. Clarkson, "Approximation algorithms for shortest path motion planning", *19STOC*, 1987, pp. 56-65.

[9] J. Choi and C.K. Yap, "Rectilinear Geodesics in 3-Space", *11 SoACM-CG*, 1995, pp. 380-389.

[10] J. Choi, J. Sellen and C.K. Yap, "Approximate Euclidean Shortest Path in 3-Space", *10 SoACM-CG*, 1994, pp. 41-48.

[11] J. Choi, J. Sellen and C.K. Yap, "Approximate Euclidean Shortest Path in 3-Space", *Intl. Jl. of Computational Geometry and Applications*,Vol. 7, no. 4, 1997, pp. 271–295.

[12] K.L. Clarkson, S. Kapoor and P.M. Vaidya, "Rectilinear Shortest Paths Through Polygonal Obstacles in O(n $\log^{3/2} n$) time", submitted for publication.

[13] 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.

[14] T. Dey, "personal communication", March 1999.

[15] E.W. Dijkstra, "A Note on Two Problems in Connection with Graphs", *Numerical Mathematics 1*, 1959, pp.269-271.

[16] M.L. Fredman and R.E. Tarjan, "Fibonacci Heaps and Their Uses in Improved Network Optimization Algorithms", *J. ACM*, **34**(3), 1987, pp.596-615.

[17] S. Har-Peled, M. Sharir, and K.R. Varadarajan, "Approximating Shortest Paths on a Convex Polytope in Three Dimensions", *12 SoACM-CG*, 1996, pp. 329-338.

[18] S. Har-Peled, "Approximate Shortest Paths and Geodesic Diameters on Convex Polytopes in Three Dimensions", *13 SoACM-CG*, 1997, pp. 359-365.

[19] J. Hershberger and J. Snoeyink, "Erased arrangements of lines and convex decompositions of polyhedra", CGTA 9, 1998, pp. 129–143.

[20] J. Hershberger and S. Suri, "Practical Methods for Approximating Shortest Paths on a Convex Polytope in $\Re^3$", *6SODA*, 1995, pp. 447-456.

[21] P. Johansson, "On a Weighted Distance Model for Injection Molding", Linköping Studies in Science and Technology, Thesis no. 604 LiU-TEK-LIC-1997:05, Applied Mathematics, Linköping University, Linköping, Sweden, 1997.

[22] C. Kenyon and R. Kenyon, "How To Take Short Cuts", *D&C Geometry*, Vol. 8, 1992, pp. 251-264.

[23] M. Lanthier, A. Maheshwari and J.-R. Sack, "Approximating Weighted Shortest Paths on Polyhedral Surfaces", *13 SoACM-CG*, 1997, pp. 274-283.

[24] M. Lanthier, “Shortest Path Problems on Polyhedral Surfaces”, *Ph.D. Thesis in progress*, School of Computer Science, Carleton University, Ottawa, Canada, 1999.

[25] M. Lanthier, A. Maheshwari, and J.-R. Sack, ”Shortest Anisotropic Paths on Terrains”, in Proc. ICALP’99, Prague, LNCS 1644, pp. 524-533, 1999.

[26] A. Maheshwari, J.-R. Sack and H. Djidjev, ”Link Distance Problems”, *Handbook on Computational Geometry*, J.-R. Sack and J. Urrutia Eds., Elsevier Science, to appear, 1999.

[27] C. Mata and J. Mitchell, “A New Algorithm for Computing Shortest Paths in Weighted Planar Subdivisions”, *13 SoACM-CG*, 1997, pp. 264-273.

[28] 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.

[29] J.S.B. Mitchell, “Shortest Paths and Networks”, *Handbook of Discrete and Computational Geometry*, J. Goodman and J. O’Rourke Eds., CRC Press, Chapter 24, 1997, pp. 445-466.

[30] J.S.B. Mitchell, “Geometric Shortest Paths and Network Optimization”, *Handbook on Computational Geometry*, J.-R. Sack and J. Urrutia Eds., Elsevier Science, to appear, 1999.

[31] C.H. Papadimitriou, “An Algorithm for Shortest Path Motion in Three Dimensions”, *IPL*, **20**, 1985, pp. 259-263.

[32] M. Sharir, “On Shortest Paths Amidst Convex Polyhedra”, *SIAM Jl. Comp.*, **16**, 1987, pp. 561-572.

[33] M. Sharir and A. Schorr, “On Shortest Paths in Polyhedral Spaces”, *SIAM J. of Comp.*, **15**, 1986, pp. 193-215.

[34] Paradigm Group Webpage, School of Computer Science, Carleton University, http://www.scs.carleton.ca/~ gis.

A Appendix

Proof of Lemma 2.2: Since each pair φ, ψ and φ_1, ψ_1 satisfies (2.1) we have

$$w''(\sin\psi_1 - \sin\psi) = w'(\sin\varphi_1 - \sin\varphi),$$

which is

$$w''\sin\frac{\psi_1-\psi}{2}\cos\frac{\psi_1+\psi}{2} = w'\sin\frac{\varphi_1-\varphi}{2}\cos\frac{\varphi_1+\varphi}{2}. \tag{1.17}$$

Using the above relation and the inequalities

$$\frac{2}{\pi}\theta \le \sin\theta \le \theta,$$

that hold for any angle $\theta \in [0, \pi/2]$ we obtain the claim of the lemma

$$|\psi_1-\psi| \le \pi\left|\sin\frac{\psi_1-\psi}{2}\right|$$
$$= \frac{\pi w'\cos\frac{\varphi_1+\varphi}{2}}{w''\cos\frac{\psi_1+\psi}{2}}\left|\sin\frac{\varphi_1-\varphi}{2}\right|$$
$$\le \frac{\pi w'}{2w''}\kappa(\varphi,\varphi_1)|\varphi_1-\varphi|.$$

Proof of Lemma 2.3: Recall that we denoted $d(e) = \sup_{x\in e} d(x)$ and let $\tau(e)$ be the ratio $d(e)/|e|$. Note that $\tau(e)$ is a positive number smaller than $1/2$ and in some sense measures the ”thickness” of the two faces around e. Let $M' \in e$ be the closest to v' point on e, such that $d(M') = d(e)$. According to our construction the point M' is between v' and M or coincides with M. Let $p'_1, \ldots, p'_{k'_1}$ be the Steiner points on the sub-segment (v', M') and $p'_{k'_1+1}, \ldots, p'_{k'}$ are those in (M', M). Then by the fact that $d(x)/|v'x|$ is a decreasing function in (v', M') we have

$$|v'p'_i| \ge (1+\varepsilon\tau(e))|v'p'_{i-1}|,$$

and therefore

$$|v'p'_i| \ge \varepsilon r(v')(1+\varepsilon\tau(e))^{i-1} \quad \text{for} \quad i = 1, \ldots, k'_1+1.$$

Since $|v'p'_{k'_1+1}| \le |v'M|$ we have

$$k'_1 \le \log_{(1+\varepsilon\tau(e))}(|v'M|/\varepsilon r(v')). \tag{1.18}$$

By the definition the points $p'_{k'_1+1}, \ldots, p'_{k'}$ are equidistantly placed in (M', M) at a distance $\varepsilon d(e)$ and therefore

$$k' - k'_1 - 1 \le \frac{|M'M|}{\varepsilon d(e)}.$$

Summing these two inequalities we obtain

$$k' - 1 \le \log_{(1+\varepsilon\tau(e))}(|v'M|/\varepsilon r(v')) + \frac{|M'M|}{\varepsilon d(e)}. \tag{1.19}$$

To estimate the number $k = k' + k'' + 1$ of Steiner points inserted in e we sum (1.19) with the analogous inequality for k'', use that $|M'M''| \le |e|$ and $4|v'M'||v''M''| \le |e|^2$ and obtain

$$\begin{aligned} k-3 & \\ \le\ & \log_{(1+\varepsilon\tau(e))}(|v'M|/\varepsilon r(v')) + \tfrac{|M'M|}{\varepsilon d(e)} \\ & + \log_{(1+\varepsilon\tau(e)}(|v''M|/\varepsilon r(v'')) + \tfrac{|M''M|}{\varepsilon d(e)} \\ \le\ & \log_{(1+\varepsilon\tau(e)}\tfrac{|v'M||v''M|}{\varepsilon^2 r(v')r(v'')} + \tfrac{|M'M''|}{\varepsilon d(e)} \\ \le\ & 2\log_{(1+\varepsilon\tau(e))}\tfrac{|e|}{2\varepsilon\sqrt{r(v')r(v'')}} + \tfrac{1}{\varepsilon\tau(e)}. \end{aligned}$$

Then the estimate stated in the lemma is derived from this inequality using properties of the logarithm function and the fact that $\varepsilon\tau(e) \le 1/2$. Namely

$$k - 3 \le 2\log_{(1+\varepsilon\tau(e))}\frac{|e|}{2\varepsilon\sqrt{r(v')r(v'')}} + \frac{1}{\varepsilon\tau(e)}$$
$$\le \frac{1}{\varepsilon\tau(e)}\left(1+\frac{5}{2}\ln\frac{|e|}{2\varepsilon\sqrt{r(v')r(v'')}}\right) \le C(e)\frac{1}{\varepsilon}\log_2\frac{2}{\varepsilon}, \tag{1.20}$$

where $C(e) < 4(|e|/d(e))\log_2(|e|/\sqrt{r(v')r(v'')})$. The estimate on the total number of Steiner points on $\mathcal{P}$ is obtained by summing (1.20) for all edges of $\mathcal{P}$.

Proof of Lemma 2.4: Let q be the point on the boundary of the union of the two faces neighboring e that minimizes the radius of the circle through p_iqp_{i+1}. Then we have $\angle p_ixp_{i+1} \le \angle p_iqp_{i+1}$. Therefore it suffices to prove (2.5) for $\angle p_iqp_{i+1}$. If we denote by ρ the radius of the circle through p_iqp_{i+1} and by θ the angle $\angle p_iqp_{i+1}$, then using (2.4) we have

$$\sin\theta = \frac{|p_{i+1}-p_i|}{2\rho} < \frac{\varepsilon\min(d(p_i),d(p_{i+1}))}{2\rho} < \varepsilon,$$

and the lemma follows.

Proof of Lemma 2.5: We first prove a). Without loss of generality assume that p_i is closer than p_{i+1} to the line containing (q_j,q_{j+1}) and similarly that q_j is closer than q_{j+1} to the line containing (p_i,p_{i+1}). It suffices to prove (2.6) for the shortest length segment joining (p_i,p_{i+1}) and (q_j,q_{j+1}). Assume that (a,b) is the shortest length segment joining (p_i,p_{i+1}) and (q_j,q_{j+1}). Then it is easily seen, that either $a=p_i$ or $b=q_j$. Furthermore, the segment (a,b) either coincides with one of the three segments (p_i,q_j), (p_i,q_{j+1}), and (p_{i+1},q_j) or (a,b) is one of the perpendiculars either from q_j to (p_i,p_{i+1}) or from p_i to (q_j,q_{j+1}).

Let us consider the case where (a,b) is one of the segments (p_i,q_j), (p_i,q_{j+1}), or (p_{i+1},q_j). By Property (2.4) we have $|p_ip_{i+1}| \le \varepsilon|ab|$ and $|q_jq_{j+1}| \le \varepsilon|ab|$. And the claim of the lemma follows from these inequalities and the triangle inequality applied appropriately as follows:

$$\text{If}\quad (a,b)=(p_i,q_j)\quad\text{then,}$$
$$|p_{i+1}q_j| \le |p_iq_j| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|;$$
$$\text{If}\quad (a,b)=(p_i,q_{j+1})\quad\text{then,}$$
$$|p_{i+1}q_{j+1}| \le |p_iq_{j+1}| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|;$$
$$\text{If}\quad (a,b)=(p_{i+1},q_j)\quad\text{then,}$$
$$|p_iq_j| \le |p_{i+1}q_j| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|.$$

Now consider the case where $b=q_j$ and (a,b) is the perpendicular from q_j to (p_i,p_{i+1}). In this case again from (2.4) and by the triangle inequality, we have

$$|p_iq_j| \le |aq_j| + |p_ia| \le |ab| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|,$$

and

$$|p_{i+1}q_j| \le |aq_j| + |ap_{i+1}| \le |ab| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|$$

which imply (2.6).

To prove (2.6) it remains to consider the case where $a=p_i$ and (a,b) is the perpendicular from p_i to (q_j,q_{j+1}). In this case the inequality $|p_iq_j| \le (1+\varepsilon)|ab|$ follows again by the triangle inequality and by (2.4), but applied to (q_j,q_{j+1}). Namely,

$$|p_iq_j| \le |p_ib| + |q_jb| \le (1+\varepsilon)|ab|.$$

The inequality

$$\min(|p_{i+1}q_j|,|p_{i+1}q_{j+1}|) \le (1+\varepsilon)|ab|$$

needs a more in-depth analysis. Denote the angle at the vertex between rays containing (p_i,p_{i+1}) and (q_j,q_{j+1}) by α and the endpoint of the perpendicular from p_{i+1} to the ray containing (q_j,q_{j+1}) by b'. If $q_{j+1}\in(b,b')$ then $|p_{i+1}q_{j+1}| \le |p_{i+1}b|$ and $|p_{i+1}b|$ can be estimated by triangle inequality as above, i.e.,

$$|p_{i+1}q_{j+1}| \le |p_{i+1}b| \le |p_ib| + |p_ip_{i+1}| \le (1+\varepsilon)|ab|,$$

that proves the claim.

Finally, we consider the case $b'\in(b,q_{j+1})$. In this case, we show that $|p_{i+1}q_j| \le (1+\varepsilon)|ab|$. Recall that $(a,b)=(p_i,b)$. Since $|p_ip_{i+1}| \le \varepsilon|ab|$, we have

$$|p_{i+1}b'| \le (1+\varepsilon\sin\alpha)|ab|.$$

Assuming $|p_{i+1}q_j| > (1+\varepsilon)|ab|$ (otherwise the claim follows) we obtain by Euclid's theorem that

$$\begin{aligned}|q_jq_{j+1}| \ge |q_jb'| &= \sqrt{|p_{i+1}q_j|^2 - |p_{i+1}b'|^2}\\ &> |ab|\sqrt{(1+\varepsilon)^2-(1+\varepsilon\sin\alpha)^2}\\ &> \varepsilon|ab|\cos\alpha.\end{aligned}$$

But this contradicts (2.4), since $d(q_j) \le |ab|\cos\alpha$. This completes the proof of (2.6).

To prove (2.7) we use twice the triangle inequality, $|p_1a| \le \varepsilon r(v)$ and (2.4) as follows

$$\begin{aligned}\max[|p_1q_j|,|p_1q_{j+1}|] &\le |p_1a| + |ab| + |q_jq_{j+1}|\\ &\le \varepsilon r(v) + (1+\varepsilon)|ab|.\end{aligned}$$

In case c) inequality (2.8) is obtained in an analogous way. The lemma has thus been proved.

Proof of Theorem 2.1:

Assume that the path $\tilde\Pi(v_0,v)$ consists of $k+2$ vertex-vicinity and $k+1$ between-vertex-vicinity portions as follows:

$$\tilde\Pi(v_0,v) = \{\tilde\Pi(S(v_0)),\tilde\Pi(S(v_0),S(v_1)),\tilde\Pi(S(v_1)),\ldots,\tilde\Pi(S(v_k),S(v)),\tilde\Pi(S(v))\} \quad (1.21)$$

We need the following definition:

Definition A.1. *A discrete path $\Pi(v_0, v)$ is a neighbour of the linear path $\tilde{\Pi}(v_0, v)$ given as above, if it is of the form*

$$\Pi(v_0, v) = \{\Pi(S(v_0)), \Pi(S(v_0), S(v_1)), \Pi(S(v_1)), \ldots, \Pi(S(v_k), S(v)), \Pi(S(v))\},$$

where its between-vertex-vicinities portions neighbour between-vertex-vicinities portions of $\tilde{\Pi}$ and its vertex-vicinity portions are two edge paths joining between-vertex-vicinities portions through the corresponding vertex.

We define a discrete path $\Pi(v_0, v)$ that neighbors $\tilde{\Pi}$ and then we will estimate its cost. Our construction follows the partition of $\tilde{\Pi}$

$$\Pi(v_0, v) = \{\Pi(S(v_0)), \Pi(S(v_0), S(v_1)), \Pi(S(v_1)), \ldots, \Pi(S(v_k), S(v)), \Pi(S(v))\}.$$

We define first the between-vertex-vicinities portions $\Pi(S(v_i), S(v_{i+1})$ of Π for $i = 0, \ldots, k$. Consider the i-th between-vertex-vicinity portion $\tilde{\Pi}(S(v_i), S(v_{i+1})$ of $\tilde{\Pi}$ and let it consist of segments $\tilde{s}_1, \ldots, \tilde{s}_l$. Denote $\tilde{s}_j = (a_{j-1}, a_j)$ for $j = 1, \ldots, l$. The bending points a_0 and a_l belong to segments (v_i, q_0) and (v_{i+1}, q_l) inside $S(v_i)$ and $S(v_{i+1})$, where q_0 and q_l are Steiner points around v_i and v_{i+1}. The other bending points $a_1, \ldots, a_{l-1}$ belong to a sequence of Steiner intervals, which we denote by by (q'_j, q''_j) for $j = 1, \ldots, l-1$. Then between-vertex-vicinities portion $\Pi(S(v_i), S(v_{i+1}))$ is a discrete path joining q_0 and q_l of the form

$$\Pi(S(v_i), S(v_{i+1})) = \{q_0, q_1, \ldots, q_l\},$$

where Steiner points q_j are defined by

$$q_j = \begin{cases} q'_j & \text{if} \quad |q_{j-1}q'_j| \le |q_{j-1}q''_j| \\ q''_j & \text{if} \quad |q_{j-1}q'_j| > |q_{j-1}q''_j| \end{cases} . \quad \text{for } j = 1, \ldots, l-1, \qquad (1.22)$$

In this way, we have defined between-vertex-vicinities portions $\Pi(S(v_i), S(v_{i+1}))$ for $i = 1, \ldots, k$ of Π. The vertex-vicinity portions are simply defined to join between-vertex-vicinities portions together. Namely, vertex-vicinity portion $\Pi(S(v_i))$ is a two segments path $\{q', v_i, q''\}$, where q' is at the end of $\Pi(S(v_{i-1}), S(v_i))$ and q'' is the beginning of $\Pi(S(v_{i-1}), S(v_i))$. The path $\Pi(v_0, v)$ is defined completely. Clearly it neighbors $\tilde{\Pi}(v_0, v)$.

Now we estimate the cost of Π. According to our construction the path Π visits consecutively the sequence of vertices $v_0, v_1, \ldots, v$. Let us estimate the sub-path $\Pi(v_i, v_{i+1})$ for some $0 \le i \le k$. This sub-path consists of the between-vertex-vicinities portion $\Pi(S(v_i), S(v_{i+1}))$ and two segments joining v_i and v_{i+1} with Steiner points on the boundary of their vertex vicinities. Therefore

$$\|\Pi(v_i, v_{i+1})\| \le \|\Pi(S(v_i), S(v_{i+1}))\| + \varepsilon(w_{\max}(v_i)r(v_i) + w_{\max}(v_{i+1})r(v_{i+1})), \qquad (1.23)$$

where $w_{\max}(\cdot)$ is the maximum weight of the faces around a vertex and $r(\cdot)$ is defined in (2.3).

Next we estimate the cost of the between-vertex-vicinities portion

$$\Pi(S(v_i), S(v_{i+1})) = \{q_0, q_1, \ldots, q_l\}.$$

The cost of the portion equals to the sum of the costs of the segments $s_j = (q_{j-1}, q_j)$ for $j = 1, \ldots, l$. These segments can be of the following three types: 1) Segments intersecting vertex vicinity (there are exactly two segments of this type – the first s_1 and the last s_l); 2) face-crossing segments; and 3) edge-using segments. We estimate the cost of the segments depending on their type. Note that the type of the segments $\tilde{s}_j$ in the original path $\tilde{\Pi}(S(v_{i-1}), S(v_i))$ matches the type of s_j.

Let $s_j = (q_{j-1}, q_j)$ for some j be a face-crossing segment. Then it approximates the face-crossing segment $\tilde{s}_j = (a_{j-1}, a_j)$ of $\tilde{\Pi}$ and according to Lemma 2.5, and Equation 2.6, we have $|s_j| \le (1+\varepsilon)|\tilde{s}_j|$. The segments s_j and $\tilde{s}_j$ belong to the same face. Therefore, the costs of traveling along them are equal and we obtain

$$\|s_j\| \le (1+\varepsilon)\|\tilde{s}_j\|. \qquad (1.24)$$

Let $s_j = (q_{j-1}, q_j)$ for some j be an edge-using segment. By our construction it approximates an edge-using segment $\tilde{s}_j = (a_{j-1}, a_j)$ of $\tilde{\Pi}$. Without lost of generality, we assume that the order of the neighboring Steiner points of a_{j-1} and a_j on the edge containing $\tilde{s}_j$ is

$$q'_{j-1} \le a_{j-1} \le q''_{j-1} \le q'_j \le a_j \le q''_j.$$

Then, the segment s_j is either (q'_{j-1}, q'_j) or (q''_{j-1}, q'_j) and thereby

$$|s_j| \le |a_{j-1}a_j| + |q'_{j-1}q''_{j-1}|. \qquad (1.25)$$

The segment $\tilde{s}_{j-1}$ must be a face-crossing segment and by (2.4), the length $|q'_{j-1}q''_{j-1}|$ is shorter than $\varepsilon|\tilde{s}_{j-1}|$. Therefore, $|s_j| \le |\tilde{s}_j| + \varepsilon|\tilde{s}_{j-1}|$. To estimate the cost $\|s_j\|$, we recall that the cost of traveling along an edge of $\mathcal{P}$ equals the minimum of the weights of the two neighboring faces and therefore the above inequality implies

$$\|s_j\| \le \|\tilde{s}_j\| + \varepsilon\|\tilde{s}_{j-1}\|. \qquad (1.26)$$

Consider now the first and the last segment in $\Pi(S(v_i), S(v_{i+1}))$. These are the segments s_0 and s_l. By Lemma 2.5, (2.8) we have

$|s_0| \le (1+\varepsilon)|\tilde{s}_0| + \varepsilon r(v_i)$ and $|s_l| \le (1+\varepsilon)|\tilde{s}_j| + \varepsilon r(v_{i+1})$, where $\tilde{s}_0$ and $\tilde{s}_l$ are the corresponding segments of $\tilde{\Pi}$ and $r(v_i)$, $r(v_{i+1})$ are the radii of the vicinities $S(v_i)$ and $S(v_{i+1})$. For the costs we have

$$\|s_0\| \leq (1+\varepsilon)\|\tilde{s}_0\| + w_{\max}(v_i)\varepsilon r(v_i), \quad (1.27)$$
$$\|s_l\| \leq (1+\varepsilon)\|\tilde{s}_l\| + w_{\max}(v_{i+1})\varepsilon r(v_{i+1}) \quad (1.28)$$

where again $w_{\max}(v_i)$ and $w_{\max}(v_{i+1})$ were defined as the maximum weights of the faces incident to v_i and v_{i+1}, respectively.

Summing the inequalities (1.24), (1.26), (1.27), and (1.28) for the segments in $\Pi(S(v_i), S(v_{i+1}))$ we obtain

$$\|\Pi(S(v_i), S(v_{i+1})\| \leq (1+2\varepsilon)\|\tilde{\Pi}(S(v_i), S(v_{i+1})\| + \varepsilon(w_{\max}(v_i)r(v_i) + w_{\max}(v_{i+1})r(v_{i+1})).$$

Now we substitute this inequality in (1.23) and obtain

$$\|\Pi(v_i, v_{i+1})\| \leq (1+2\varepsilon)\|\tilde{\Pi}(S(v_i), S(v_{i+1})\| + 2\varepsilon(w_{\max}(v_i)r(v_i) + w_{\max}(v_{i+1})r(v_{i+1})) \quad (1.29)$$

By the formula (2.3) we have

$$w_{\max}(v_i)r(v_i) + w_{\max}(v_{i+1})r(v_{i+1}) = \frac{1}{5}(w_{\min}(v_i)d(v_i) + w_{\min}(v_{i+1})d(v_{i+1})). \quad (1.30)$$

On the other hand, Definition 2.1 and (2.3) implies that

$$\frac{4}{5}w_{\min}(v_i)d(v_i) \leq \|\tilde{\Pi}(S(v_i), S(v_{i+1}))\|. \quad (1.31)$$

We sum (1.31) with its symmetric version for v_{i+1} and obtain

$$\frac{1}{5}(w_{\min}(v_i)d(v_i) + w_{\min}(v_{i+1})d(v_{i+1})) \leq \frac{1}{2}\|\tilde{\Pi}(S(v_i), S(v_{i+1}))\|.$$

By substitution in (1.30) and (1.29) we have

$$\|\Pi(v_i, v_{i+1})\| \leq (1+3\varepsilon)\|\tilde{\Pi}(S(v_i), S(v_{i+1}))\|.$$

The theorem is obtained by summing the above inequality for $i = 0, \ldots, k$.

Proof of Lemma 2.6: Let us denote by a_1, a_2, b_1, b_2, and c_1, c_2 the left and right Steiner points neighboring a, b and c respectively. By the definition Π neighbors a geodesic path $\tilde{\Pi}$ and let the bending points of $\tilde{\Pi}$ sharing common Steiner intervals with a, b and c be $\tilde{a}$, $\tilde{b}$ and $\tilde{c}$. Thereby we have $\tilde{a} \in (a_1, a_2)$, $\tilde{b} \in (b_1, b_2)$, and $\tilde{c} \in (c_1, c_2)$.

The segment (a, b) is either a face-crossing or an edge-using segment. We consider these two possibilities separately.

Case 1. Let (a, b) be a face-crossing segment.
First we shall estimate the differences between in-angles of the segments (a, b) and $(\tilde{a}, \tilde{b})$. Let us denote these two in angles by φ and $\tilde{\varphi}$. Recall that φ and $\tilde{\varphi}$ are clockwise angles between segments (a, b) and $(\tilde{a}, \tilde{b})$ and normals at b and $\tilde{b}$. We are going to show that

$$|\varphi - \tilde{\varphi}| \leq \pi\varepsilon. \quad (1.32)$$

One easily observes that $|\varphi - \tilde{\varphi}|$ always equals the angle θ between lines containing (a, b) and $(\tilde{a}, \tilde{b})$ and so it is enough to show $\theta \leq \pi\varepsilon$. Let the intersection point between lines containing $(a, b)$and $(\tilde{a}, \tilde{b})$ by x (if these lines do not intersect θ is zero and the claim is trivial). Then

$$\theta = \angle ax\tilde{a} \leq \angle ab\tilde{a} + \angle \tilde{b}\tilde{a}b,$$

where we have equality if the point $x = (a, b) \cap (\tilde{a}, \tilde{b})$ and inequality if (a, b) and $(\tilde{a}, \tilde{b})$ do not intersect. Now it follows that $\theta \leq \pi\varepsilon$, since by Lemma 2.4 both $\angle ab\tilde{a}$ and $\angle \tilde{b}\tilde{a}b$ are at most $\pi\varepsilon/2$.

The same argument implies that

$$|\psi_c - \tilde{\psi}| \leq \pi\varepsilon, \quad (1.33)$$

where ψ_c and $\tilde{\psi}$ are the out-angles of the segments (b, c) and $(\tilde{b}, \tilde{c})$.

Using that the angle $\angle cbc_0$ is equal to $|\psi - \psi_c|$ triangle inequality and (1.33) we obtain

$$\angle cbc_0 = |\psi - \psi_c| \leq |\psi_c - \tilde{\psi}| + |\psi - \tilde{\psi}| \leq \pi\varepsilon + |\psi - \tilde{\psi}|. \quad (1.34)$$

Next we estimate $|\psi - \tilde{\psi}|$. Lemma 2.2 and (1.32) imply

$$|\psi - \tilde{\psi}| \leq \frac{\pi w'}{2w''}\kappa(\varphi, \tilde{\varphi})\pi\varepsilon$$

and by using a simple inequality $\kappa(\varphi, \tilde{\varphi}) \leq \pi/(\pi - |\psi + \tilde{\psi}|)$ we have

$$|\psi - \tilde{\psi}| \leq \frac{\pi^3 w'\varepsilon}{2w''(\pi - |\psi + \tilde{\psi}|)}. \quad (1.35)$$

On the other hand, since $\psi, \tilde{\psi} \in [-\pi/2, \pi/2]$ we have $|\psi - \tilde{\psi}| + |\psi + \tilde{\psi}| \leq \pi$ and therefore

$$|\psi - \tilde{\psi}| \leq \min\left(\pi - |\psi + \tilde{\psi}|, \frac{\pi^3 w'\varepsilon}{2w''(\pi - |\psi + \tilde{\psi}|)}\right) \leq \pi\sqrt{\frac{\pi w'\varepsilon}{2w''}}. \quad (1.36)$$

We obtain the lemma in the case where (a, b) is a face-crossing segment by substitution of (1.36) in (1.34).

Case 2. Let (a, b) be an edge-using segment.
In this case by Lemma 2.1 we know that the out-angles ψ and $\tilde{\psi}$ are equal. Thus

$$\angle cbc_0 = |\psi - \psi_c| = |\tilde{\psi} - \psi_c|.$$

But as we have proved $|\tilde{\psi} - \psi_c| < \pi\varepsilon$ and the lemma follows.

Proof of Lemma 2.7 : We choose (b, c') to be the edge of the spanner that lie in the same sub-cone as (b, c). Then the inequality $|bc'| \leq |bc|$ follows by the definition of a spanner and $|bc| \leq |bc'| + |cc'|$ is the triangle inequality. To prove that the last inequality we use that for any triangle bcc' we have

$$|bc'| + |cc'| \leq (1 + 2\frac{\sin(\angle b/2)\sin(\angle c/2)}{\sin(\angle c'/2)})|bc|,$$

where $\angle b$, $\angle c$, and $\angle c'$ denote the angles at b, c and c' respectively. Additionally we have that

$$\sin(\angle b/2) \leq \angle b/2 \leq \pi\varepsilon/4$$

and $|bc'| < |bc|$ implies $\sin(\angle c/2) \leq \sin(\angle c'/2)$. The inequality we need follows.

Complete description of the set $Cone^*(\Pi, a_{k-1})$in Section 2.4: The following cases arise for the last node and edge of the path $\Pi = \{a_0, a_1, \ldots, a_{k-1},\}$:

***Case 1*:** The node a_{k-1} is a vertex of $\mathcal{P}$.
In this case the set $Cone^*(\Pi, a_{k-1})$ consists of all edges incident to a_{k-1} except (a_{k-2}, a_{k-1}). The number of edges in $Cone^*(\Pi, a_{k-1})$ in this case is equal to the degree of the vertex a_{k-1} in $\mathcal{P}$.

***Case 2*:** The node a_{k-1} neighbours a vertex of $\mathcal{P}$.
In this case the edges incident to a_{k-1} can be partitioned into two cones – one per each of the two faces incident to a_{k-1}. Then the set $Cone^*(\Pi, a_{k-1})$ is defined as the union of the ε-spanners of these cones. The number of edges in $Cone^*(\Pi, a_{k-1})$ in this case does not exceed $\lceil 4/\varepsilon \rceil$.

***Case 3*:** The node a_{k-1} neither neighbours nor is a vertex of $\mathcal{P}$.
In this case the definition of $Cone^*(\Pi, a_{k-1})$ depends on the type of the segment (a_{k-2}, a_{k-1}) and we consider the two possible sub-cases.

***Case 3.1*:** The edge (a_{k-2}, a_{k-1}) is an edge-using segment.
In this case the set $Cone^*(\Pi, a_{k-1})$ depends on the first node a_{k-i}, $i > 2$ backward on the path Π, for which (a_{k-i}, a_{k-1}) is a face-crossing segment (see Figure 6). The set $Cone^*(\Pi, a_{k-1})$ consists of the edge joining a_{k-1} with its neighboring Steiner point different of a_{k-2} plus ε-spanners of the geodesic cones of the segments (a_{k-i}, a_{k-1}) and (a_{k-i+1}, a_{k-1}). Note that in some of the cases the corresponding geodesic cones are empty and so are their spanners.

To estimate the size of the set $Cone^*(\Pi, a_{k-1})$ in this case we use the estimates for the angles of the geodesic cones obtained in Lemma 2.6. Then we have estimates on the size of the spanners based on the size of the cone angles. We add these estimates, use that $\varepsilon < 1$ and obtain

$$|Cone^*(\Pi, a_{k-1})| \leq 2 + \pi^2(2 + \sqrt{\frac{\pi w'}{2w''}})\varepsilon^{-\frac{1}{2}}. \tag{1.37}$$

Here w' is the weight of the face containing (a_{k-i}, a_{k-1}) and w'' is the weight of the other face incident to a_{k-1}.

***Case 3.2*:** The edge (a_{k-2}, a_{k-1}) is a face-crossing segment.
In this case the set $Cone^*(\Pi, a_{k-1})$ consists of the two edges joining a_{k-1} with its neighboring Steiner points plus the ε-spanner of the geodesic cone $GCone(a_{k-2}, a_{k-1})$. The angle of the geodesic cone was estimated in Lemma 2.6 and according to the construction of ε-spanner and (2.9) the size of $Cone^*(\Pi, a_{k-1})$ is at most

$$|Cone^*(\Pi, a_{k-1})| \leq 3 + \pi^2(1 + \sqrt{\frac{\pi w'}{2w''}})\varepsilon^{-\frac{1}{2}}, \tag{1.38}$$

where w' is the weight of the face containing (a_{k-2}, a_{k-1}) and w'' is the weight of the other face incident to a_{k-1}.

Proof of Lemma 2.8: We prove the lemma by induction on k. If $k = 1$, i.e. the path $\Pi(a_0, a_1)$ consists of a single edge (a_0, a_1), then according to our definition $\Pi(a_0, a_1) \in \mathcal{C}^*(a_0)$ and the claim follows. Assume now that $k > 1$ and that there is a path $\Pi^*(a_0, a_{k-1}) \in \mathcal{C}^*(a_0)$ with the structure described in the lemma and such that

$$\|\Pi^*(a_0, a_{k-1})\| \leq (1 + \pi\varepsilon/2)\|\Pi(a_0, a_{k-1})\|.$$

We will show that the path $\Pi^*(a_0, a_{k-1})$ can be continued to a_k satisfying the requirements of the lemma. By Definition 2.5 the possible one-edge continuations of $\Pi^*(a_0, a_{k-1})$ form a set of edges $Cone^*(\Pi^*, a_{k-1})$, that was described in cases 1–3 above.

Observe first that if the edge (a_{k-1}, a_k) belongs to $Cone^*(\Pi^*, a_{k-1})$ then the lemma immediately follows for the path

$$\Pi^*(a_0, a_k) = \Pi^*(a_0, a_{k-1}) \cup (a_{k-1}, a_k).$$

This is always valid if the edge (a_{k-1}, a_k) joins two neighboring Steiner points. Thus we assume below that (a_{k-1}, a_k) is a face-crossing segment.

We consider cases corresponding to Cases 1–3 above with respect to the path $\Pi^*(a_0, a_{k-1})$.

1. In the case where a_{k-1} is a vertex of $\mathcal{P}$) the edge (a_{k-1}, a_k) belongs to $Cone^*(\Pi^*, a_{k-1})$ and the lemma follows.

2. In the case where a_{k-1} neighbors a vertex of $\mathcal{P}$, the set $Cone^*(\Pi^*, a_{k-1})$ consists of the two ε-spanners covering all the edges incident to a_{k-1}. Therefore either $(a_{k-1}, a_k) \in Cone^*(\Pi^*, a_{k-1})$ or there is an edge (a_{k-1}, a'_k) in $Cone^*(\Pi^*, a_{k-1})$ so that (a'_k, a_k) is an edge-using segment and (Lemma 2.11)

$$|a_{k-1}, a'_k| \leq |a_{k-1}, a_k| \leq |a_{k-1}, a'_k| + |a'_k, a_k| \leq (1 + \pi\varepsilon/2)|a_{k-1}, a_k|. \quad (1.39)$$

Then we consider the path

$$\Pi^*(a_0, a_k) = \Pi^*(a_0, a_{k-1}) \cup \{a_{k-1}, a'_k, \ldots, a_k\},$$

that goes along the edge-using segment (a'_k, a_k) visiting the corresponding sequence of Steiner points. This path belongs to the class $\mathcal{C}^*(a_0)$ and has the required structure. The estimate on the cost of this path follows from the inductive assumption and the inequality

$$\|\{a_{k-1}, a'_k, \ldots, a_k\}\| \leq (1 + \pi\varepsilon/2)\|a_{k-1}, a_k\|,$$

which is true because of (1.39).

3. Assume now that a_{k-1} neither neighbors nor is a vertex of $\mathcal{P}$.

3.2. In the case (Case 3.2) where the last edge of $\Pi^*(a_0, a_{k-1})$ is a face-crossing segment, we know by the inductive assumption that this segment is exactly (a_{k-2}, a_{k-1}). We assumed that the segment (a_{k-1}, a_k) is a face crossing segment and thus it belongs to the geodesic cone $GCone(a_{k-2}, a_{k-1})$. By our definition (Case 3.2), the set $Cone^*(\Pi^*, a_{k-1})$ contains an ε-spanner of $GCone(a_{k-2}, a_{k-1})$ and the lemma follows by the same argument as in the case 2 above.

3.1 It remains to consider the case where the last edge in $\Pi^*(a_0, a_{k-1})$ joins two neighboring Steiner points. By our inductive assumption on the structure of the path $\Pi^*(a_0, a_{k-1})$ we have that

$$\Pi^*(a_0, a_{k-1}) = \Pi^*(a_0, a_{k-i}) \cup \{a'_{k-i+1}, \ldots, a_{k-i+1}\} \cup \{a_{k-i+1}, \ldots, a_{k-1}\},$$

where (a_{k-i}, a_{k-i+1}) for some $i > 1$ is the first face-crossing edge backward on the path Π. Thus the vertex determining the set $Cone^*(\Pi^*, a_{k-1})$ is a_{k-i}.

If $i = 2$ then (a_{k-2}, a_{k-1}) is a face-crossing edge and (a_{k-1}, a_k) belongs to the geodesic cone $GCone(a_{k-2}, a_{k-1})$, whose ε-spanner is in $Cone^*(\Pi^*, a_{k-1})$ and the lemma follows by the argument presented for the case 2.

If $i > 2$ then (a_{k-2}, a_{k-1}) is an edge-using segment and (a_{k-1}, a_k) belongs to the geodesic cone $GCone(a_{k-i+1}, a_{k-1})$. But $Cone^*(\Pi^*, a_{k-1})$ contains a ε-spanner of this cone too and the lemma is proved completely.

Algorithm PrunedDijkstra:
Input: The surface $\mathcal{P}$, the set of Steiner points V_ε and a fixed Steiner point a_0.
Output: The output tree $SPT(\mathcal{C}^*(a_0))$ is a set pointers $parent(a)$ for each $a \in V_\varepsilon$.
Step 1. (*Initialization*)
1.1 For each node $a \in V_\varepsilon$ do
$key(a) = \infty$
1.2 Set $key(a_0) = 0$ and $parent(a_0) = a_0$
1.3 Initialize a priority queue $Q = V_\varepsilon$
Step 2. (*Construction of* $SPT(\mathcal{C}^*(a_0))$)
While $Q \neq empty$ do
2.1 $a = Extract - min(Q)$
2.2 For each edge $(a, b) \in Cone^*(\Pi(a), a)$ do
If $key(b) > key(a) + \|(a, b)\|$ then
$Decrease - key(Q; key(b) = key(a) + \|(a, b)\|)$
$parent(b) = a$

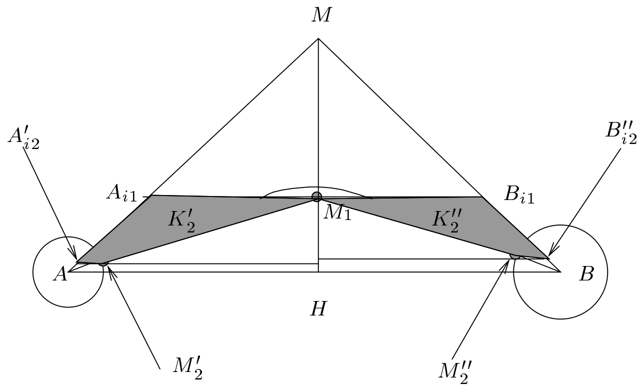
Proof of Lemma 2.9 with reference to Algorithm PrunedDijkstra: The first part of the lemma can be derived analogously to the corresponding part of Dijkstra's correctness proof. The time complexity follows from Lemma 2.3, where we established $|V_\varepsilon| = O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon})$.

Proof of Lemma 3.1: We shall estimate the number of Steiner points placed in the triangle ABM. We denote $h = p_0 = |MH|$ and $p_i = |P_i H|$ for $i = 1, 2, \ldots$. In this notation we have

$$p_{i-1} - p_i = \varepsilon d(P_i) = p_i \varepsilon \sin\gamma$$
$$\text{and thereby} \quad p_i = h(1 + \varepsilon \sin\gamma)^{-i}, \quad (1.40)$$

where γ is the smallest of the two dihedral angles incident to f at AB. Also according to this notation and (3.14) the number k_i is

$$k_i = \left\lceil \frac{|A_i, B_i|}{\varepsilon p_{i+1} \sin\gamma} \right\rceil = \left\lceil \frac{|A_i, B_i|(1 + \varepsilon \sin\gamma)^{i+1}}{h\varepsilon \sin\gamma} \right\rceil. \quad (1.41)$$



Let i_1 be the smallest index such that the segment $A_{i_1}B_{i_1}$ is at distance smaller than $\varepsilon r(e)$ from AB. Let us denote by K_1 the number of Steiner points lying on segments A_iB_i with $i_1 > i$ and by K_2 the number of the remaining Steiner points.

The number K_1 can be estimated as follows

$$K_1 = \sum_{i=1}^{i_1-1} k_i = \sum_{i=1}^{i_1-1} \left\lceil \frac{|A_i, B_i|}{\varepsilon p_{i+1} \sin\gamma} \right\rceil \leq$$
$$i_1 - 1 + \frac{|AB|}{(h\varepsilon \sin\gamma)^2}(1+\varepsilon\sin\gamma)^{i_1+1}. \qquad (1.42)$$

Using the definition of the index i_1 and (1.40) we derive

$$i_1 = \left\lceil \log_{(1+\varepsilon\sin\gamma)} \frac{h}{\varepsilon r(e)} \right\rceil, \qquad (1.43)$$

which after substitution in (1.42) gives

$$K_1 \leq \log_{(1+\varepsilon\sin\gamma)} \frac{h}{\varepsilon r(e)} + \frac{|AB|}{\varepsilon^3 h r(e) \sin^2\gamma}(1+\varepsilon\sin\gamma)^2 \qquad (1.44)$$

Next we estimate K_2, that is the number of Steiner points lying on segments A_iB_i with $i \geq i_1$. By our definitions on the segment $A_{i_1}B_{i_1}$ there is a point M_1 such that the triangle ABM_1 lies entirely inside $\mathcal{V}(AB)$, (Figure A). Let M'_2 and M''_2 be the points on AM_1 and BM_1 respectively for which

$$|AM'_2| = \varepsilon r(A) \quad \text{and} \quad |BM''_2| = \varepsilon r(B).$$

Let i'_2 be the smallest index such that the segment $A_{i'_2}B_{i'_2}$ is closer to AB than M'_2. Similarly, let i''_2 be the smallest index so that the segment $A_{i''_2}B_{i''_2}$ is closer to AB than M''_2. All Steiner points on segments A_iB_i with $i > i_1$ lie in the trapezoids $A_{i_1}A_{i'_2}M'_2M_1$ and $B_{i_1}B_{i''_2}M''_2M_1$. We denote the numbers of Steiner points in these two trapezoids by K'_2 and K''_2 respectively.

To estimate the number K'_2 of the Steiner points in $A_{i_1}A_{i'_2}M'_2M_1$ we show an upper bound on the number of the Steiner points on A_iB_i and lying inside this trapezoid. Namely, if we denote this number by $k'(i)$ and by M_i the intersection point between A_iB_i and AM_1 for $i = i_1, \ldots, i'_2$ we have

$$k'(i) - 1 \leq \frac{|A_iM_i|}{\varepsilon p_{i+1} \sin\gamma} = \frac{|A_{i_1}M_1|}{h\varepsilon\sin\gamma}(1+\varepsilon\sin\gamma)^{i_1+1}.$$

Thus for the number of Steiner points inside the trapezoid $A_{i_1}A_{i'_2}M'_2M_1$ we have

$$K'_2 \leq (i'_2 - i_1)\left(1 + \frac{|A_{i_1}M_1|}{h\varepsilon\sin\gamma}(1+\varepsilon\sin\gamma)^{i_1+1}\right).$$

Analogously, for the number of Steiner points inside the trapezoid $B_{i_1}B_{i''_2}M''_2M_1$ we obtain

$$K''_2 \leq (i''_2 - i_1)\left(1 + \frac{|B_{i_1}M_1|}{h\varepsilon\sin\gamma}(1+\varepsilon\sin\gamma)^{i_1+1}\right),$$

and summing this two and using (1.43) we obtain

$$K_2 = K'_2 + K''_2 \leq (i'_2 + i''_2 - 2i_1)(1 + \frac{|AB|}{\varepsilon^2 r(e)\sin\gamma}(1+\varepsilon\sin\gamma)^2). \qquad (1.45)$$

¿From the definitions of the indices i'_2, i''_2 and (1.40) we easily derive

$$i'_2 \leq 1 + \log_{(1+\varepsilon\sin\gamma)} \frac{h|AB|}{\varepsilon^2 r(e) r(A)},$$
$$i''_2 \leq 1 + \log_{(1+\varepsilon\sin\gamma)} \frac{h|AB|}{\varepsilon^2 r(e) r(B)}.$$

Then using (1.43) we estimate

$$i'_2 + i''_2 - 2i_1 = 2 + 2\log_{(1+\varepsilon\sin\gamma)} \frac{|AB|}{\varepsilon\sqrt{r(A)r(B)}}. \qquad (1.46)$$

Finally, combining (1.44), (1.45), (1.46) and using properties of the logarithm function we obtain an estimate on the total number of Steiner points in the triangle ABM

$$K_1 + K_2 \leq C_{ABM}(\mathcal{D})\frac{1}{\varepsilon^3}\ln\frac{1}{\varepsilon}, \qquad (1.47)$$

where the constant $C_{ABM}(\mathcal{D})$ depends on the geometry of the two tetrahedra around the triangle ABM and also logarithmically on the maximum ratio of the weights around vertices A and B. Readily the estimate (1.47) applies for the number of Steiner points in the triangle ABC with a constant $C_{ABC}(\mathcal{D})$ that is the sum of the three constants for the triangles defined by M. □