A polynomial time algorithm for finding an approximate shortest path amid weighted regions

Rajasekhar Inkulu∗ Sanjiv Kapoor†

Abstract

We devise a polynomial-time approximation scheme for the classical geometric problem of finding an $\epsilon$-short path amid weighted regions. In this problem, a triangulated region $P$ comprising of $n$ triangles, a positive weight associated with each triangle, and two points $s$ and $t$ that belong to $P$ are given as the input. The objective is to find a path whose cost is at most $(1 + \epsilon)OPT$ where $OPT$ is the cost of an optimal path between $s$ and $t$. Our algorithm initiates a discretized-Dijkstra wavefront from source $s$ and progresses the wavefront till it strikes $t$. This result is about a cubic factor (in $n$) improvement over the Mitchell and Papadimitriou ’91 result [8], which is the only known polynomial time algorithm for this problem to date.

1 Introduction

The problem of computing a shortest path in polygonal subdivisions is well-studied due to its applications in geographic information systems, VLSI design, robot motion planning, etc., Mitchell [7] surveyed various shortest path related problems and results. In this paper, we devise an algorithm for the weighted shortest path problem [8]: given a triangulation $P$ with $n$ faces, each face associated with a positive weight, finding a path between two input points $s$ and $t$ (both belonging to $P$) so that the path has minimum cost among all possible paths joining $s$ and $t$ that lie on $P$. The cost of any path $p$ is the sum of costs of all line segments in $p$, whereas the cost of a line segment is its Euclidean length multiplied by the weight of the face on which it lie. The weighted shortest path problem helps in modeling region specific constraints in planning motion.

To compare the time complexities of various algorithms in the literature, we use the following notation - $n$: number of vertices defining $P$; $L$: length of the longest edge bounding any face of $P$; $N$: maximum coordinate value used in describing $P$; $w_{\text{max}}$: maximum non-infinite weight associated with any triangle; $w_{\text{min}}$: minimum weight associated with any triangle; $\theta_{\text{min}}$: minimum among the internal face angles of $P$; and, $\mu$: ratio of $w_{\text{max}}$ to $w_{\text{min}}$. (Note that the same notation is also used in later parts of the paper.)

Mitchell and Papadimitriou [8] presented an algorithm that finds an approximate shortest path in $O(n^8 \log \frac{nN^{\mu}}{\epsilon})$. Their algorithm essentially builds a shortest path map for $s$ by progressing continuous-Dijkstra wavefront in $P$ using the Snell’s law of refraction. By introducing $m$ equi-spaced Steiner points on each edge of $P$ and building a graph spanner over these points, Mata and Michell [9] devised a preprocessing algorithm to construct a graph spanner in $O(kn^3)$ time, where $k = O(\frac{\mu}{w_{\text{min}}})$, on which $(1 + \epsilon)$-approximate shortest path queries are performed. Lanthier et al. [5] independently devised a $O(n^5)$ time approximation algorithm with an additive error of $O(Lw_{\text{max}})$ by choosing $m = n^2$. Instead of uniform discretization (as in [6]), Aleksandrov et al. [2, 3] used logarithmic discretization and devised a $O(n^2 \log \frac{N^{\mu}}{w_{\text{max}}}) \log \frac{1}{\epsilon}$ time approximation algorithm. Sun and Reif [9] provided an approximation algorithm, popularly known as BUSHWHACK, with time complexity $O(n^2 \log \frac{N^{\mu}}{w_{\text{max}}}) \log \frac{1}{\epsilon}$. Their algorithm dynamically maintains for each Steiner point $v$, a small set of incident edges of $v$ that may contribute to an approximate shortest path from $s$.

∗Dept. of Computer Science and Engineering, IIT Guwahati, India. rinkulu@iitg.ac.in
†Dept. of Computer Science, IIT Chicago, USA. kapoor@iit.edu
to $t$. The query versions’ of this problem are addressed in [2, 3, 9, 8, 3, 11]. Further, algorithms in Cheng et al. [1] handle more generic case of measuring the cost of path length in each face with an asymmetric convex distance function.

The time complexities of all the above mentioned solutions, except for [8], are polynomial in parameters such as $\epsilon, \theta_{\text{min}}, N, \mu$. Since these parameters depend on the input size, strictly speaking, these algorithms are not polynomial. Like [8], this paper devises an algorithm that is polynomial in time complexity. The time complexity of our algorithm is $O(n^5 \text{ poly-log})$, which is about a cubic factor improvement from [8]. As established in [8], the lower bound of the problem stands at $\Omega(n^4)$. Further, [8] also bounds the number of event points required to define the interaction of the shortest path map with $\mathcal{P}$ as $\Omega(n^4)$. Our algorithm takes first steps to provide a sub-quadratic (as a function on the number of events) solution.

This result uses several of the characterizations from [8], and extends their solution to obtain better bounds on the time complexity. Our approach discretizes the wavefront in the continuous-Dijkstra’s approach by a set $S$ of rays whose origin is source $s$. These rays are distributed uniformly around $s$. As the discrete wavefront propagates, a subset of the rays in $S$ are progressed further while following the Snell’s laws of refraction. Each of these subsets of rays is guided by two extreme rays from that subset; all the rays in that subset lie between these two special rays. Essentially, each such subset represents a section of the wavefront. For any vertex $v$ in $\mathcal{P}$, whenever such a section of the wavefront strikes any close-by point to $v$, we initiate another discrete-wavefront (set of rays) from $v$. We continue doing this until the wavefront (approximately) strikes $t$.

Apart from finding a shortest path from $s \in \mathcal{P}$ to each vertex $v$ of $\mathcal{P}$, with minor modifications, our solution can extend to find all-pairs shortest paths. Further, our algorithm is also applicable in finding geodesic shortest paths on the surface of a 2-manifold whose faces are associated with positive weights.

Section [2] lists relevant propositions from [8] and [9], and defines terminology required to describe the algorithm. Section [3] outlines the algorithm while introducing few structures used in the algorithm. In Section [4] we bound the number of rays. The details of the algorithm are provided in Section [5]. Section [6] analyzes the algorithm. Further, conclusions are given in Section [7].

2 Preliminaries

A path is a continuous image of an interval, say $[0, 1]$, in the plane. A path $p$ is said to cross an edge $e$ whenever there is a point of intersection between $p$ and the interior of $e$. A geodesic path is a path that is locally optimal and cannot, therefore, be shortened by slight perturbations. An optimal path is a geodesic path that is globally optimal. The general form of a weighted geodesic path is a simple (that is, not self-intersecting) piece-wise linear path that goes through zero or more vertices while possibly crossing a zero or more edges. We denote the weight of a face $f$ (resp. edge $e$) with $w_f$ (resp. $w_e$). The Euclidean length of a line segment $l$ is denoted with $\|l\|$. Let $p$ be a geodesic path with line segments $l_1, l_2, \ldots, l_k$ such that $l_i$ lies on face $f_i$, for every $i$ in $[1, k]$, then the weighted Euclidean length (a.k.a. the cost) of path $p$ is defined as the $\sum_i \|l_i\| w_{f_i}$.

The shortest path problem is: given a finite triangulation $\mathcal{P}$ in the plane with two points $s$ (source) and $t$ (destination) located on $\mathcal{P}$, an assignment of positive integral weights to faces of $\mathcal{P}$, and an error tolerance $\epsilon$, find a path $p$ from $s$ to $t$ that lie on $\mathcal{P}$ such that the cost of $p$ is at most $(1 + \epsilon)$ multiplication factor away from the cost of an optimal path from $s$ to $t$. We assume that all parameters of the problem are specified by integers. In particular, all vertices have non-negative integer coordinates. Further, we assume that $s$ and $t$ are two vertices of $\mathcal{P}$. For an edge $e$ is shared by faces $f'$ and $f''$, the weight of $e$ is defined as $\min(w_{f'}, w_{f''})$.

Let $a$ be a point in $\mathbb{R}^2$. Let $r_1, r_2$ be two rays that originate from $a$ and make angles $\theta'$ and $\theta''$, with respect to a reference line. A cone $C(a, \theta', \theta'')$ is the area swept by $r_1$ when $r_1$ is rotated in counterclockwise direction till it coincides with $r_2$.

2.1 Few facts from literature

First we list few propositions, definitions, and descriptions from Mitchell and Papadimitriou [8] tailored for our purpose. A sequence of edge-adjacent faces is a list, $(f_1, f_2, \ldots, f_{k+1})$, of two or
more faces such that, for every \( i \), face \( f_i \) shares edge \( e_i \) with face \( f_{i+1} \). Further, the corresponding sequence of edges \( \mathcal{E} = (e_1, e_2, \ldots, e_k) \) is referred as an edge sequence. When a geodesic path \( p \) crosses edges in \( \mathcal{E} \) in the order specified by \( \mathcal{E} \) and without passing through any vertex, then \( \mathcal{E} \) is said to be the edge sequence of path \( p \).

Let \( f' \) and \( f'' \) be two faces with shared edge \( e \). Let \( p \) be a geodesic path and let \( r', r'' \) be two successive line segments along \( p \) with \( r' \) lying on \( f' \) and \( r'' \) lying on \( f'' \). Also, let \( \theta' \) (resp. \( \theta'' \)) be the counterclockwise angle between the incoming ray \( r' \) (resp. \( r'' \)) and the normal to \( e \), known as the angle of incidence of \( r' \) onto \( e \) (resp. angle of refraction of \( r'' \) from \( e \)). When \( w_{f'} > w_{f''} \), the critical angle of \( e \), denoted with \( \theta_c(f', f'') \), is \( \sin^{-1}(\frac{w_{f''}}{w_{f'}}) \). If \( \theta' < \theta_c(f', f'') \), then the angles \( \theta' \) and \( \theta'' \) are related by the Snell’s law of refraction with \( w_{f'} \sin \theta' = w_{f''} \sin \theta'' \). Given that no geodesic path incidents to an edge \( e \) from \( f' \) at an angle greater than \( \theta_c(f', f'') \), the only other possibility is \( \theta' \) being equal to \( \theta_c(f', f'') \). Let \( y' \) be the point of incidence of \( r' \) onto \( e \) with angle of incidence \( \theta_c(f', f'') \). Then the geodesic path travels along \( e \) for some positive distance (in the direction of its orientation), and then exits edge \( e \) back into face \( f' \), say at a point \( y'' \) located in the interior of \( e \), leaving the edge \( e \) at an angle \( -\theta_c(f', f'') \). We say that the path is critically reflected by edge \( e \) and the line segment \( y'y'' \) of \( e \) is a critical segment of \( p \) on \( e \). The point \( y' \) (the closer of the two points \( \{y', y''\} \) to \( s \)) along \( p \) is known as a critical point of entry of path \( p \) from face \( f' \) and \( y'' \) is known as the corresponding critical point of exit of path \( p \) into face \( f'' \).

Lemmas 3.7, 7.1, and 7.4 from [8] are useful for our algorithm and hence respectively listed herewith.

**Proposition 1** Let \( p \) be a geodesic path. Then either (i) between any two consecutive vertices on \( p \), there is at most one critical point of entry to an edge \( e \), and at most one critical point of exit from an edge \( e' \) (possibly equal to \( e \)); or (ii) the path can be modified in such a way that case (i) holds without altering the length of the path.

A locally \( f \)-free path to a point \( x \in e \) strikes \( x \) from the exterior of face \( f' \) and is locally optimal.

**Proposition 2** Let \( p \) be a shortest locally \( f \)-free path. Let \( p' \) be a sub-path of \( p \) such that \( p' \) goes through no vertices or critical points. Then, \( p' \) can cross an edge \( e \) at most \( O(n) \) times. Thus, in particular, the length of any edge sequence of path \( p \) is \( O(n^2) \).

**Proposition 3** Any shortest geodesic path \( p \), passes through \( O(n) \) critical points of entry on any given edge \( e \). Therefore, the number of critical points of entry along a shortest geodesic path are \( O(n^2) \).

The following proposition on non-crossing property of shortest paths is from Sun and Reif [9].

**Proposition 4** Any two shortest geodesic paths with the same source point cannot intersect in the interior of any region.

### 3 Algorithm outline

As propagating the continuous-Dijkstra wavefront exactly is complicated, we approximate the wavefront by discretizing the wavefront with rays. Every ray is a geodesic path on \( P \). We initiate a set \( \mathcal{R}(s) \) of rays whose origin is \( s \). The rays in \( \mathcal{R}(s) \) are ordered according to their counterclockwise angle with the x-axis and are uniformly distributed around \( s \). We define two rays \( r', r'' \) in \( \mathcal{R}(s) \) to be successive whenever there is no ray from \( \mathcal{R}(s) \) that lies between \( r' \) and \( r'' \). The number of rays in \( \mathcal{R}(s) \) is defined by the angle \( \delta \) between successive rays; by expressing the value of \( \delta \) in terms of \( \epsilon \) later (in Section 3), our algorithm ensures \( \epsilon \)-approximation. These rays together are described as a discrete wavefront initiated at \( s \). When the wavefront strikes any vertex \( v \) for the first time, analogously, an ordered set \( \mathcal{R}(v) \) of rays are initiated from \( v \), unless vertex \( v \) is the destination \( t \) itself.
3.1 Types of rays

A ray initiated is progressed further only if necessary; a ray that is considered for propagation is said to have been traced. We progress the rays as described below. Let \( e \) be a common edge between faces \( f' \) and \( f'' \). When a ray \( r \in \mathcal{R}(v) \) is traced along face \( f' \), suppose it strikes \( e \) at a point \( y \in e \). Based on the weights \( w_{f'}, w_{f''} \), and the angle of incidence of \( r \) onto \( e \), either of the following may happen:

(i) angle of incidence of \( r \) onto \( e \) is less than \( \theta_c(f', f'') \)

(ii) angle of incidence of \( r \) onto \( e \) is greater than \( \theta_c(f', f'') \)

(iii) angle of incidence of \( r \) onto \( e \) is equal to \( \theta_c(f', f'') \)

In case (i), \( r \) refracts into face \( f'' \) with the angle of refraction defined by Snell’s law of refraction. In case (ii), since \( w_{f'} > w_{f''} \), we do not progress \( r \) any further as \( r \) does not cause a shortest path to \( t \). In case (iii), \( r \) travels along \( e \) for a positive distance before critically reflecting back into face \( f'' \) itself. Let \( \kappa \) be the corresponding critical segment. Since the shortest path can be reflected back from any point on \( \kappa \), the algorithm will generate rays that originate from various points on \( \kappa \). The number and positions of points from which these rays are generated is again a function of \( e \). These rays reflect back into face \( f' \) with the angle of exit from \( e \) equal to \(-\theta_c(f', f'')\). We let \( \mathcal{R}(\kappa) \) be the ordered set of rays that originate from \( \kappa \), and are ordered by the distance along \( \kappa \) from \( y \).

To account for the divergence of rays, we further initiate an ordered set of Steiner rays, denoted with \( \mathcal{R}(y) \), from each critical point of entry \( y \). Hence, a critical point of entry is also termed as a critical source. See Fig. 4. These rays are motivated as follows: a pair of successive rays when progressing along an edge sequence can diverge non-uniformly, especially at angles close to the critical angles. To establish the approximation, instead of having a more denser set of rays from every source, we fill the gaps by generating Steiner rays from critical points of entries. This helps in reducing the overall time complexity. Let \( v' \) and \( v'' \) be the endpoints of \( e \). Also, let the critical segment \( \kappa \) due to ray \( r \) be \( yv'' \). As a result of discretization, there may not exist rays in \( \mathcal{R}(v) \) that spread in the cone \( C(y, \frac{\pi}{2}, \theta_c(f', f'')) \) in the face \( f'' \). (The angles are measured with respect to the normal to \( e \) in the appropriate direction.) To account for this gap, an ordered set \( \mathcal{R}(y) \) of rays are initiated from \( y \) that lie in the cone \( C \), ordered by the counterclockwise angle each ray makes with the normal to \( e \). Every ray in \( \mathcal{R}(y) \) is said to be a Steiner ray.

To recapitulate, there are three kinds of rays:

- \( \{ \mathcal{R}(v) \mid v \in \mathcal{P} \} \)
- \( \{ \mathcal{R}(\kappa) \mid \text{a ray } r \in \mathcal{R}(v) \text{ for some } v \in \mathcal{P} \text{ caused the critical segment } \kappa \} \)
- \( \{ \mathcal{R}(y) \mid y \text{ is a critical point of entry for a ray } r \in \mathcal{R}(v) \text{ for some } v \in \mathcal{P} \} \)

We sometimes denote the origin of a ray \( r \) with \( \text{origin}(r) \); the \( \text{origin}(r) \) could either be a vertex, a critical source, or a critical segment.

3.2 Ray bundles

Since we intend to approximate the continuous-Dijkstra wavefront, instead of tracing individual rays, we trace rays in pairs so that a section of the continuous-Dijkstra wavefront is assumed to be in-between two such rays. The rays could be initiated from either of these: a vertex \( v \), a critical source \( c \), or a critical segment \( \kappa \). Let \( o \) be one such source of rays. Let \( B \) be a maximal set of rays in \( \mathcal{R}(o) \) such that any two rays in \( \mathcal{R}(o) \) cross the same edge sequence \( \mathcal{E} \) when traced for the current progress state of the discrete wavefront. Then \( B \) is said to be a ray bundle of \( \mathcal{R}(o) \). Further, \( \mathcal{E} \) is said to be the edge sequence associated with \( B \); also, \( \mathcal{E} \) is said to be associated with any ray in \( B \). Let \( r' \) and \( r'' \) be two rays in \( B \) such that there exists a point on \( r' \) and a point on \( r'' \) wherein the line segment joining them intersects every other ray from \( B \) (when traced). Then the rays \( r' \) and \( r'' \) are said to be the sibling pair of ray bundle \( B \) and rays in \( B \) are said to be siblings. Further, instead of tracing all the rays in the ray bundle \( B \), we trace the sibling pair of \( B \) only; this helps in reducing the time complexity.
The sibling rays and the ray bundles are updated as the rays progress further. For the sibling pair \( r', r'' \) of ray bundle \( B \), when the edge sequence of \( r' \) changes from \( r'' \) for the first time, by doing binary search among the rays in \( B \) we trace the appropriate rays across the edge sequence of \( B \) and form new sibling pairs. Let \( abc \) be a face \( f \) in \( T \). Suppose rays \( r' \) and \( r'' \) are siblings in bundle \( B \) till they strike edge \( ab \) of \( f \). However, when they are traced further, suppose \( r' \) strikes edge \( ac \) and \( r'' \) strikes edge \( bc \). At this instance of wavefront progression, the edge sequences of \( r' \) and \( r'' \) differ for the first time, and hence \( r' \) and \( r'' \) do not belong to the same bundle from there on. With binary search over the rays in \( B \), a pair of successive rays, say \( r'_1 \) and \( r''_1 \), are found to make new sibling pairs: \( r', r'_1 \) and \( r'', r''_1 \). Further, the ray bundle \( B \) is split accordingly. We say (sibling) rays \( r'_1 \) and \( r''_1 \) struck the vertex \( c \). We also term that the ray bundle has been split. The details of the algorithm to find rays \( r'_1 \) and \( r''_1 \) is described later.

The significance of tracing rays versus initiating rays remains same with sets of rays in \( R(y) \) and \( R(\kappa) \), where \( y \) is a critical point of entry and \( \kappa \) is a critical segment. The rays in \( R(\kappa) \) are distinct in the sense that the paths of any two rays in \( R(\kappa) \) are parallel between any two successive edges in the edge sequence associated with the corresponding ray bundle.

### 3.3 Tree of rays

Let \( v \) be a vertex in \( P \). Let \( S_1 \) be the set of critical sources such that for any source \( q \) in \( S_1 \), a discrete wavefront is initiated from \( q \) when the discrete wavefront that originated at \( v \) strikes \( q \). Also, let \( S_1, S_2, \ldots, S_{i-1}, S_i, \ldots, S_k \) be the sets of critical sources such that for every \( q \in S_i \), critical source \( q \) is created due to the critical incidence of some ray originated from a source in \( S_{i-1} \) onto \( q \). We organize sources \( \{ v \} \cup (\cup_j S_j) \) into a tree of rays, denoted with \( T_R(v) \). The set of nodes of \( T_R(v) \) comprises of \( \{ v \} \cup (\cup_j S_j) \): more specifically, \( v \) is the root node; and, every other node in \( T_R(v) \) is a distinct critical source from \( \cup_j S_j \). For any two nodes \( u \in S_{i-1}, w \in S_i \) in \( T_R(v) \), \( u \) is the parent of \( w \) in \( T_R(v) \) whenever \( w \) is struck for the first time by a sibling pair wherein at least one ray of that pair is originated from \( u \).

We generalize sibling pair and ray bundle notions further. For any two rays \( r \) and \( r' \) belonging to \( T_R(v) \), rays \( r \) and \( r' \) are siblings whenever the suffix of edge sequence associated with \( r \) is same as the edge sequence associated with \( r' \), or vice versa. A maximal set \( S \) of rays initiated from the nodes of \( T_R(v) \) that are siblings to each other is a ray bundle. In a ray bundle \( B_v \), two rays, say \( r' \) and \( r'' \) are said to be the sibling pair of \( B_v \) whenever there exist a point \( q' \in r' \) and \( q'' \in r'' \) such that the line segment \( q'q'' \) intersects every ray in \( B_v \).

Therefore, there are three kinds of sibling pairs possible: (i) both the rays in a sibling pair originate from the same vertex, (ii) both the rays in a sibling pair originate from the same critical segment, and, (iii) sources of both the rays in a sibling pair belong to a tree of rays. Ray bundles corresponding to (iii) are also qualified: a ray bundle of \( T_R(v) \). To distinguish, siblings of the first two kinds with the last, the latter may be qualified by saying a sibling pair in \( T_R(v) \).

Let \( r', r'' \) be a sibling pair of \( T_R(v) \). Let \( u, w \in T_R(v) \) be the respective origins of \( r' \) and \( r'' \). Also, let \( v' \) be the least common ancestor of \( u \) and \( w \) in \( T_R(v) \). The path \( p \) in tree \( T_R(v) \) from \( v' \) to \( u \) comprising of an alternating sequence of critical sources and critically incident rays is the critical ancestor path of \( r' \) with respect to \( r'' \). Note that the critical ancestor path of a ray is always defined with respect to a sibling pair, and more importantly in reference to a tree of rays. As described later, the critical ancestor path of a ray helps in efficiently splitting a sibling pair of \( T_R(v) \).

### 4 Bounding the number of initiated and traced rays

Before detailing the algorithm, we bound the number of both the initiated and traced rays.

The shortest path from the source \( s \) to the destination can be split into sub-paths where each sub-path can be classified into the following types:

- A Type-I path that does not use any critical segment
- A Type-II path that uses at least one critical segment
The following theorems consider these types of paths in both ensuring the \( \epsilon \)-approximation together with the time complexity.

**Theorem 4.1** For any vertex \( u \) in \( \mathcal{P} \), let \( \frac{\theta'}{2\pi} (\frac{\epsilon}{K})^3 \) be the maximum angle between any pair of successive rays in \( \mathcal{T}_{R}(u) \), wherein \( K \) is a large constant and \( \epsilon = \frac{\epsilon}{n^3(1+\sin_{\max})} \). Then a Type-1 shortest path from any vertex \( u \) to any point in \( \mathcal{P} \) can be \( \epsilon \)-approximated using rays in \( \mathcal{T}_{R}(u) \).

To prove the above theorem, we proceed as follows: given a Type-1 path \( P \) from a vertex, say \( u \), to another point \( v \in \mathcal{P} \), we show that there exists a Type-1 path \( P' \) using rays in our discretization so that \( P' \) closely approximates \( P \). We establish bounds on the error while using a sequence of vertices \( w_i, i = 1 \ldots k \), and rays in \( \mathcal{T}_{R}(w_i) \) to determine a shortest path.

**Lemma 4.1** Let \( r', r'' \) be a sibling pair of \( \mathcal{T}_{R}(u) \). Also, let \( q' \) and \( q'' \) be two points on \( r' \) and \( r'' \), respectively, at weighted Euclidean distance \( d \) from \( s \) and lying on the same face. If the angle \( \delta \) between \( r' \) and \( r'' \) is less than or equal to \( \frac{1}{2\pi} (\frac{\epsilon}{\sin_{\max}})^3 \epsilon' \), then the weighted Euclidean distance of the line segment joining \( q' \) and \( q'' \) is upper bounded by \( d \epsilon' \). (The parameter \( \beta \) is defined in the proof.)

**Proof:**

![Diagram illustrating the construction of Lemma 4.1](image)

(The distances' shown in the figure are Euclidean (i.e., unweighted).)

We first show a bound on the angle between a pair of successive rays so as to limit the divergence of the rays in \( \mathcal{T}_{R}(u) \). Let the sibling pair \( r', r'' \) traverse across regions with weights \( w_1, w_2, \ldots, w_p \) respectively. Let \( e_1, e_2, \ldots, e_{p-1} \) be the edge sequence associated with this sibling pair. W.l.o.g., we consider the case when the angle between the rays \( r' \) and \( r'' \), is increasing due to refractions at these edges; and, for the worst-case divergence to occur we assume that \( W = w_1 \geq w_2 \geq \ldots \geq w_p = w \geq 1 \). Let \( \theta_1', \theta_2', \ldots, \theta_{p-1}' \) be the angles at which the ray \( r' \) incidents on edges \( e_1, e_2, \ldots, e_{p-1} \) respectively. Let \( \theta_2, \theta_3, \ldots, \theta_p \) be the angles at which the ray \( r'' \) refracts at edges \( e_1, e_2, \ldots, e_{p-1} \) respectively. Similarly, let \( \theta_1' + \delta_1', \theta_2' + \delta_2', \ldots, \theta_{p-1}' + \delta_{p-1}' \) be the angles at which the ray \( r'' \) incidents on edges \( e_1, e_2, \ldots, e_{p-1} \) respectively. And, let \( \theta_2 + \delta_2, \theta_3 + \delta_3, \ldots, \theta_p + \delta_p \) be the angles at which the ray \( r'' \) refracts at edges \( e_1, e_2, \ldots, e_{p-1} \) respectively. See Fig. 1. Assume that \( \theta_p \leq \theta_{\text{critical}} \leq \frac{\pi}{2} \); and, \( v, i, \delta_i \) is small.

For every integer \( i \in [1, p-1] \),

\[
 w_i \sin \theta_i' = w_{i+1} \sin \theta_{i+1}
\]  

(1)

For every integer \( i \in [1, p-1] \),

\[
 w_i \sin (\theta_i' + \delta_i') = w_{i+1} \sin (\theta_{i+1} + \delta_{i+1})
\]
Since $\delta'_i$ and $\delta_{i+1}$ are small, we approximate $\sin \delta_i, \cos \delta_i, \sin \delta_{i+1},$ and $\cos \delta_{i+1}$ with the first two terms of the Maclaurin series:

\[
\sin \theta'_i + w_i \delta_i \cos \theta'_i = w_{i+1} \sin \theta_{i+1} + w_{i+1} \delta_{i+1} \cos \theta_{i+1}
\]
\[
\cos \theta'_i = w_{i+1} \delta_{i+1} \cos \theta_{i+1} + \theta'(1) \Rightarrow \delta_{i+1} = \frac{w_{i+1}}{w_i} \frac{\cos \theta'_i}{\cos \theta_{i+1}} \leq \frac{w_{i+1}}{w_i} \frac{\cos \theta'_i}{\cos \theta_{i+1} - \frac{\delta_{i+1}}{2} \sin \theta_{i+1} - \frac{\delta_{i+1}}{6} \cos \theta_{i+1}} \delta'_i
\]

(2)

Therefore, $\delta_p \leq \frac{w_1}{w_p} \left( \max_1 \frac{\cos \theta'_i}{|\cos \theta_{i+1} - \frac{\delta_{i+1}}{2} \sin \theta_{i+1} - \frac{\delta_{i+1}}{6} \cos \theta_{i+1}|} \right)^p \delta'_i$

Letting $\max_1 \leq \beta$, the above becomes

\[
\delta_p \leq \frac{w_1}{w_p} \beta^p \delta'_i
\]

(3)

Note that for the case in which $p_3$ and $p_4$ lie in two neighboring regions, there exists a vertex between the rays $p_1p_3$ and $p_1p_4$, hence this proof holds good. We next consider the distance between two points, both on the same face $f$, such that these points (i) lie on a pair of successive rays $r'$ and $r''$ in $T_R(u)$, and (ii) are at equal weighted Euclidean distance from $u$. Let $p_1$ be the point of incidence of $r''$ on an edge $e_i$ with an angle of incidence $\theta''_i$, and let it refract from $p_1$ at an angle of refraction $\theta_{i+1}$. Further, let the ray $r''$ incident on $e_i$ at $p'_2$ with an angle of incidence $\theta''_i + \delta'_i$ and refract from $p'_2$ at an angle of refraction $\theta_{i+1} + \delta_{i+1}$, See Fig. 1. Consider two points $p_1, p_2$ located on rays $r', r''$ respectively such that $p_1, p_2$ are in the face $f_i$, with weight $w_i$, and the weighted Euclidean distance from $u$ to either of these points is $d$. Let $\|p_1p_2\| = d_1$. Assume w.l.o.g. that $r''$ is incident to $e_i$ at a larger angle than $r'$. Also, consider two points $p_3, p_4$ located on rays $r', r''$ respectively such that $p_3, p_4$ are in the region $f_{i+1}$ with weight $w_{i+1}$, the weighted Euclidean distance from $u$ to either of these points being $d'$, and $\|p_3p_4\| = d_2$. We wish to establish a bound on $d_2$. Note that it suffices to restrict attention to the above case when both $p_3$ and $p_4$ are in $f_{i+1}$. If one of the points, say $p_4$, is in $f_i$ then the bound on $d_2$ will be smaller since $w_{i+1} \leq w_i$. Also, let $\|p_2p'_2\| = d'_1, \|p'_2p'_4\| = d'_2$. We choose $d''_i$ such that $d''_i \sin (\theta_{i+1} + \delta_{i+1}) = d''_i \sin (\theta''_i + \delta'_i)$,

\[
(d' - d) = w_i d'_1 + w_{i+1} d'_2 = \frac{w_i d''_i \sin (\theta_{i+1} + \delta_{i+1})}{\sin (\theta''_i + \delta'_i)} + w_{i+1} d'_2 = \frac{w_i d''_i}{w_{i+1}} + w_{i+1} d'_2
\]

(4)

We assume that for $p_1$ and $p_2$, which are at a weighted Euclidean distance $d$ from $u$, the following is true:

\[
w_i d_1 \leq d'
\]

(5)

for some $\epsilon' < \epsilon$; we intend to find conditions for

\[
w_{i+1} d_2 \leq d' \epsilon'
\]

(6)
and, for $0 \leq (\theta'_i + \delta_i) \leq (\theta'_{i+1} + \delta_{i+1}) \leq \pi/2$,\sin(\theta'_i + \delta_i) \leq \sin(\theta'_{i+1} + \delta_{i+1}))$
$\leq |d' + (d' - d)\delta_{i+1}| + |(d' - d)\delta_{i+1}|$ (from [5], and, for small $\delta_{i+1}$)
$\leq d' + (d' - d)(2\delta_{i+1})$.

Since $\delta_{i+1} \leq \delta_p$, and, from [5], $2\delta_{i+1} \leq 2\delta_p \leq 2\frac{w}{w_p} \beta_p \delta'_1 \leq 2\frac{w}{w_p} \beta^2 \delta'_1$ (in the worst-case, $p$ the number of edges intersected by a ray is $O(n^2)$).
To satisfy [5], we need to have, $2\frac{w}{w_p} \beta^2 \delta'_1 \leq \epsilon'$ i.e., $\delta'_1 \leq \frac{1}{2} \frac{w}{\beta} (\frac{1}{\beta} \gamma)^2 \epsilon'$.
We thus get the claimed bound on $\delta$.

\[ \square \]

**Lemma 4.2** Let $u$ and $v$ be vertices in $\mathcal{P}$ such that the shortest distance from $u$ to $v$ via a Type-1 path is $d(u, v)$. The distance from $u$ to $v$ via a traced ray in $\mathcal{T}_R(u)$ is less than $d(u, v)(1 + \epsilon' \mu(1 + \frac{1}{\sin \theta_{min}}))$.

![Figure 2: Illustrating the construction described in Lemma 4.2](image)

**Proof:** We let $\mathcal{E}$ be the sequence of edges traversed by an optimum path from $u$ to $v$. Let $v$ be the vertex to which edges $e_i$ and $e_j$ incident. Let $e_k$ be the last edge in $\mathcal{E}$, after which the shortest path enters triangle $e_i, e_j, e_k$, which corresponds to face $f$.

Consider a sibling pair $r_i, r_j \in \mathcal{R}(u)$ whose associated edge sequence is $\mathcal{E}$ such that the ray $r_i$ is incident to edge $e_i$ at $p_i$, closest to $v$ and the ray $r_j$ is incident to edge $e_j$ at $p_j$, again closest to $v$. See Fig. 2. Suppose $\|v p_i\| = l_i$ and $\|v p_j\| = l_j$. Let $p_i$ be at weighted Euclidean distance $d_i$ from $u$ and let $p_j$ be at weighted Euclidean distance $d_j$ from $u$. W.l.o.g., assume that $d_i \leq d_j$. Let $p'_j$ be the point on ray $r_j$ at Euclidean distance $l_j$ from $p_i$. Then from Lemma 4.1 the distance between $p_i$ and $p'_j$ is at most $d_i \epsilon'$. Thus the weighted Euclidean distance from $u$ to $v$ via $p_i$ is upper bounded by $d_i + w_{e_i} l_i$, where $w_{e_i}$ is the weight along the edge $e_i$. Further, $l_i \leq (d_i \epsilon' / w_f + (d - d_i) / w_f)$, where $w_f$ is the weight of the face $f$. Let $d_i \epsilon' / w_f = x$ and let $(d - d_i) / w_f = y$. Then $x \geq y \sin \theta_{min}$ or $y \leq x / (\sin \theta_{min})$. Thus $l_i \leq d_i \epsilon'(1 + \frac{1}{\sin \theta_{min}}) / w_f$. Thus the distance from $u$ to $v$ via $p_i$ is as claimed.

\[ \square \]

![Figure 3: Illustrating the construction of theorem 4.2](image)

The above lemmas will be used to bound the approximation of the distance from $u$ to any point that lies between the two successive rays in $\mathcal{P}$. Now consider the case in which two successive rays, one of which refracts and the other causes a critical source. Consider two such successive rays $r_i$ and $r_j$ originated from the same point, say $v_1$. See Fig. 3. Let $r_i$ be a ray that incident to a point $p''_1$ located on edge $e$, at an angle less than the critical angle. Also, let $r_j$ be a ray that incident to a point $p_1$ located on edge $e$, at an angle equal to the critical angle of edge $e$. Let $d_P(x, y)$ represent the distance between points $x$ and $y$ along path $P$. Suppose an optimal shortest path $P_{OPT}$ intersects edge $e$ between $p''_1$ and $p_1$ at $p'_1$. Let $P_{OPT}$ refract at point $p'_1$ at an angle $\theta_{opt}$ with the normal. Consider a path $P$ that approximates $P_{OPT}$ and that uses ray $r_j$ from $v_1$ to $p_1$
and then along a path that is the shortest path from \( p_1 \) to \( v_2 \). Then the modified path has length specified by
\[
d_P(v_1, v_2) \leq d_P(v_1, p_1) + d_O(p_1, v_2)
\]
where \( d_O(p_1, v_2) \) is the shortest path from \( p_1 \) to \( v_2 \) in the discretized space. Note that \( d_P(v_1, p_1) \leq d_{OPT}(v_1, p_1') + d(p_1', p_1) \). Let \( p_1' \) be at weighted Euclidean distance \( d_1 \) from \( v_1 \) and, let \( p_1 \) be at weighted Euclidean distance \( d_1 \) from \( v_1 \). By an analysis similar to that in the previous lemma, \( d_P(v_1, p_1') \leq d_{OPT}(v_1, p_1')(1 + \epsilon'(\mu(1 + \frac{1}{\sin \theta_{\min}}))) \). Further, \( d_P(p_1', v_2) \leq d_{OPT}(p_1', v_2)(1 + \epsilon'(\mu(1 + \frac{1}{\sin \theta_{\min}}))) \). Thus \( d_P(v_1, v_2) \leq d_{OPT}(v_1, v_2)(1 + \mu(1 + \frac{1}{\sin \theta_{\min}})\epsilon') \). Since there can be at most \( O(n^2) \) critical points of entry along any path, on repeating the above analysis for each approximation by a ray from a critical point of entry will result in an error factor of \( O(n^2\mu(1 + \frac{1}{\sin \theta_{\min}})\epsilon') \).

Given that there are \( O(n) \) vertices on a shortest path and the successive errors can accumulate, the total error in computing the optimal distance of value \( d \) from \( u \) to \( v \) is at most \( O(d\epsilon'(n^2\mu(1 + \frac{1}{\sin \theta_{\min}}))) \). Since this should not exceed \( d\epsilon \), we choose \( \epsilon' = \frac{\epsilon}{n(n^2\mu(1 + \frac{1}{\sin \theta_{\min}}))} \). Note that the approximation is achieved by rays which are closest to the vertex \( v \). These rays are a sibling pair in \( R(u) \). This completes the proof of Theorem 4.1.

**Theorem 4.2** Let \( P \) be a Type-2 shortest path from a vertex (or, a critical source) \( v \) to a vertex \( w \) on \( P \) such that \( P \) uses a critical segment \( \kappa \). If the angle between a pair of successive rays is as specified in Theorem 4.1 then an \( \epsilon \)-approximate Type-2 shortest path can be found from a pair of successive rays in \( R(v) \), for any \( v \) that is either a vertex or a critical source.

**Proof:** The proof of Lemma 4.2 shows that a sibling pair in \( R(v) \) and sibling rays originating from \( \kappa \) will determine the approximate shortest path. An optimal path \( P \) can be partitioned into two, a path \( P_1 \) from \( v \) to \( \kappa \) and a path \( P_2 \) from \( \kappa \) to \( w \). Let \( e \) be the edge on a face \( f \) such that the critical segment \( \kappa \) lies on \( e \) and reflects rays back onto face \( f \). Also, let the critical segment \( \kappa \) have a critical point of entry \( y \). We first show that a good approximation to the path from \( \kappa \) to \( w \) can be found by showing that sibling rays are correctly maintained. In fact, we need to find the sibling rays closest to vertex \( w \) when the rays from \( \kappa \) strikes edges incident to \( w \). Note that initially, two rays from the end points of \( \kappa \) are generated onto face \( f \) correctly as the sibling rays. Subsequently, as these rays strike edges of the faces they are refracted and siblings updated. If the rays strike different edges \( e_1 \) and \( e_2 \), \( e_1 \cap e_2 = w \), after striking an edge \( e' \) of a face comprising edges \( e', e_1 \) and \( e_2 \), the appropriate siblings are computed. Also the shortest distance path to \( w \) starting at the critical segment \( \kappa \) is computed. This path has a first segment that is parallel to the sibling rays, \( r' \) and \( r'' \), and maintained during the progress of the propagation. Furthermore it has its origin, \( p \), same as the origin of \( r \), located in-between the origins of \( r' \) and \( r'' \). If the sibling rays do not split but strike one edge, say \( e_1 \), the distance via the closest sibling to \( w \) is computed as required in Lemma 4.2. Finally, a Type-1 path between \( u \) and \( y \) is found correctly by the algorithm as specified in Lemma 4.2. \( \square \)

## 5 Details of the algorithm

The algorithm is event driven, where the events are considered in their weighted Euclidean distance from \( s \). The algorithm starts with initiating a set \( R(s) \) of rays that are uniformly distributed around \( s \). The various types of events that need to be both determined and handled are described in the following subsections.

### 5.1 Initiating rays from a vertex

This procedure is invoked to initiate rays from a given vertex, say \( v \), when the discrete wavefront strikes \( v \). Since \( s \) is also a vertex of \( P \), this procedure is also used in initiating rays from \( s \) as well. Let \( f' \) be the face along which a pair of successive rays are traced to strike \( v \). Let \( F(v) \) be the collection of all the faces incident to \( v \) except for \( f' \). The set \( R(v) \) of rays are initiated from \( v \), and all these rays lie on the set \( F(v) \) of faces. Due to Proposition 4.3 (non-crossing property of weighted
shortest paths), we do not initiate rays from $v$ over the face $f'$. Further, the angle between any two successive rays in $\mathcal{R}(v)$ at $v$ are bounded by $\delta$ (whose value is bounded in Section 4).

![Illustrating successive rays (shown in blue color) striking $v$ from $f'$ and a ray bundle getting initiated on a face $f \in \mathcal{F}(v)$; sibling pair of $B_f$ is shown in red. The sibling pair shown on $f''$ is shown in green color; when $r'$ along $f$ and/or $r'''$ along $f''$ strike $v'v''$ and $v'v'''$ edges respectively, a discrete wavefront is initiated from $v'$.]

Consider any face $f \in \mathcal{F}(v)$. Let $vv', vv'', v'v'''$ be the edges bounding $f$. See Fig. 4. The set $B_f \subseteq \mathcal{R}(v)$ of rays that lie on face $f$ is a ray bundle: every ray in $B_f$ strikes $v'v''$ before striking any other edge in $\mathcal{P}$. For every such ray bundle $B_f$, we find a sibling pair corresponding to $B_f$. Let $r'$ (resp. $r''$) be the ray in $B_f$ such that there does not exist a ray in $B_f$ between $r'$ (resp. $r''$) and $vv'$ (resp. $vv''$). We do binary search (with respect to edges $vv'$ and $vv''$) among the rays in $\mathcal{R}(v)$ to find both the rays $r'$ and $r''$. Let $q_{v'}$ and $q_{v''}$ be the points on edge $v'v''$ to which rays $r'$ and $r''$ incident when they are traced. The following sets of event points are pushed to the event heap: the event point corresponding to tracing the ray $r'$ (resp. $r''$) to $q_{v'}$ (resp. $q_{v''}$) that occurs at the weighted Euclidean distance $d_v + w_f||vq_{r'||}||$ (resp. $d_v + w_f||vq_{r'''}||$) from $s$. Here, $d_v$ is the weighted Euclidean distance between $v$ and $s$ i.e., when the discrete wavefront struck $v$. Further, the corresponding sibling pairs is saved with each event point.

Let $f, f''$ be two adjacent faces in $\mathcal{F}(v)$. See Fig. 4. Let $vv''v'''$ (resp. $vv'v'''$) be the triangle defining $f$ (resp. $f''$). Let $r'$ be the ray in $\mathcal{R}(v)$ that lies on $f$ and let $r'''$ be the ray in $\mathcal{R}(v)$ that lies on $f''$ such that no ray in $\mathcal{R}(v)$ lies between $r'$ and $r'''$. Also, let $q_{r'}$ (resp. $q_{r'''}$) be the point on edge $v'v''$ (resp. $v'v'''$) to which ray $r'$ (resp. $r'''$) incident when traced. Then the event point for initiating a discrete wavefront from $v'$ is pushed to the event heap with the key value $\min(d_v + w_f||vq_{r'}|| + w_{v'v''}||q_{r'}v'||, d_v + w_f||vq_{r'''}|| + w_{v'v'''}||q_{r'''}v'''||)$.

To improve the time complexity, we use the non-crossing property of shortest paths (Proposition 4): whenever a ray bundle $B$ originated at a vertex $v'$ strikes another vertex $v$, we save that information with $v$ so that whenever another ray bundle $B'$ originated from the same vertex $v'$ strikes $v$, we do not progress $B'$ any further. We do the relevant book-keeping to achieve this.

### 5.2 Handling the ray striking an edge critically

Let $r_1, r_2$ be a sibling pair. Let $e(v', v'')$ be a common edge to faces $f$ and $f'$. Consider the following event point: when a ray $r_1$ is traced along face $f$, let it critically incident to $e$ at a point $y \in e$. When this event occurs, we initiate two kinds of rays: set $\mathcal{R}(\kappa)$ of rays originated from the critical segment $yv''$ (denoted with $\kappa$); set $\mathcal{R}(y)$ of Steiner rays originated from the critical source $y$. In the following subsections, we describe algorithms to both initiate these sets of rays and to set up the ray bundles.

#### 5.2.1 Initiating rays from a critical segment

We describe the procedure to initiate rays from the critical segment $\kappa$ first. Considering a geodesic shortest path can be reflected back onto $f$ from any point on $\kappa$, we initiate a discrete wavefront from $\kappa$. The initiated rays reflect back into $f$ with the angle of exit from $e$ equal to $-\theta_c(f, f')$. Note that the rays in the discrete wavefront are parallel to each other; further, to achieve $\epsilon$-approximation, distance $\delta$ between any two such successive rays along $e$ is upper bounded by $\epsilon'$ (where $\epsilon'$ is defined in Theorem 1).

Let $y_1$ be the point located on $e$ at $\|yv''\| - \epsilon'$ Euclidean distance from $y$ such that $y_1$ is located between $y$ and $v''$. Let $r$ and $r'$ be two critically reflected (parallel) rays (making an angle $-\theta_c(f, f')$ with the normal to $e$), originated from points $y$ and $y_1$ respectively.
If both the rays incident to same edge, say \( e'' \) of \( f \), we set \( r, r' \) as a sibling pair and all the rays between \( r \) and \( r' \) that cross the same edge sequence together with \( r, r' \) is a ray bundle. See Fig. 5. Let \( x \) (resp. \( x'' \)) be the point at which the ray \( r \) (resp. \( r' \)) incidents onto edge \( e'' \). This sibling pair is pushed to the event heap: so that the ray \( r \) strikes \( x \) at the weighted Euclidean distance \( d_y + w_f \|yx\| \) from \( s \); and the ray \( r' \) strikes \( x'' \) at the weighted Euclidean distance \( d_{y1} + w_f \|y_1x''\| \).

If the rays in a sibling pair \( r, r' \) incident to distinct edges of \( f \), we need to find two rays \( r'_1, r'_2 \) in \( R(κ) \), to respectively pair up with \( r \) and \( r' \), for forming two new sibling pairs (in turn, ray bundles). See Fig. 6. Further, we also need to store an event point in the event heap that correspond to initiating a discrete wavefront from \( v \). The algorithm to split a sibling pair originated from a critical segment is mentioned in subsection 5.4.1.

### 5.2.2 Initiating rays from a critical source

There are three cases to consider based on the origin of the sibling rays that initiated the discrete wavefront from the critical source \( y \):

(i) \( r_1, r_2 \) are originated from some critical segment

(ii) \( r_1, r_2 \) are originated from some vertex

(iii) \( r_1, r_2 \) are originated from (possibly distinct) nodes of a tree of rays, say \( T_R(w) \)

We know from Proposition 1 that along a geodesic path, a critical point of exit and a critical point of entry cannot occur in succession, hence case (i) cannot occur. Since case (iii) is a generalization of case (ii), herewith we explain event handling for case (iii).

Let \( u \in T_R(w) \) be the origin of \( r_1 \). Note that \( u \) may be \( w \) itself or a distinct critical source in \( T_R(w) \). When they are distinct, a node corresponding to critical source \( y \) is inserted as a child of \( u \) in \( T_R(w) \); further, a set \( R(y) \) of rays are initiated from \( y \). The rays in \( R(y) \) are uniformly distributed in the cone \( C(y, \frac{ε'}{K}, θ') \); here, \( θ' \) is the clockwise angle ray \( r_2 \) makes with the normal to critical segment and \( \frac{ε'}{K} \) is the counterclockwise angle with respect to critical segment. Both the \( ε' \) and \( K \) are defined in Section 4. See Fig. 7. A ray bundle and a sibling pair corresponding to that ray bundle are determined: Let \( r_3 \) be the ray in \( R(y) \) that makes minimum angle with \( yv'' \). Then \( r_2, r_3 \) is set as a sibling pair of \( T_R(w) \). The ray \( r_3 \) is interpreted as a ray that originates from the origin of ray \( r_1 \). This sibling pair is traced over the face \( f'' \).
When traced, if both \( r_2 \) and \( r_3 \) incident to same edge of face \( f' \), then they together will continue to be a sibling pair. Otherwise, to form two sibling pairs, new rays to pair up with \( r_2 \) and \( r_3 \) are found from the sets of rays initiated in \( T_R(w) \); the procedure is detailed in Subsection 5.4.2. These sibling pairs’ are pushed to the event heap with the key value being the respective weighted Euclidean distances’ from \( s \) to the points at which these pairs strike the edges of \( f' \).

5.3 Extending rays across a region

Given the bundles of rays that strike an edge \( e = (a, b) \), the rays need to be extended across the region they enter. Let \( T = (a, b, c) \) be the triangular region. The rays that refract after striking \( e \), as well as the critical rays that originate from \( e \), need to be extended to determine their strike points on the other two edges \((a, c)\) and \((b, c)\).

Extending a bundle involves splitting a sibling pair. While the split operation details will defined subsequently, we discuss how to manage the extension of the set of bundles. Note that only one of the bundles that strike \((a, b)\), will lead to the shortest path from the origin \( s \) to the vertex \( c \) via edge \((a, b)\). Furthermore,

(i) From a pair of bundles that cross each other, only one will be retained, due to the non-crossing property of shortest paths.

(ii) From a pair of bundles that split at vertex \( c \) only one bundle need be retained to be split, the one that determines the shortest distance to \( c \).

Determining the shortest path to \( c \) via rays contained in a bundle \( B \) will be detailed later. Given the shortest path to \( c \) via each bundle in the current set of bundles, \( \mathcal{B} \) that strikes \( e \), one can determine the shortest path to \( c \) w.r.t \( \mathcal{B} \). The bundle \( B' \in \mathcal{B} \) that determines the shortest path is maintained. Note that \( \mathcal{B} \) changes as bundles strike edge \( e \), but there are only \( O(n) \) bundles in total. The bundle \( B' \) partitions the bundles in \( \mathcal{B} \) into two sets of bundles that strike either \((a, c)\) or \((b, c)\). These bundles are extended after processing their strike on their corresponding edge. To ensure that only relevant bundles in \( \mathcal{B} \) are extended from an edge, say \( e = (a, c) \) we determine the shortest path to the edge \( e \) via rays in bundle \( B \in \mathcal{B} \). If this shortest distance determines the overall shortest distance to the corresponding point on the edge \( e \), then the bundle is useful and may be extended across the edge \( e \). The shortest path to the edge can be determined by a binary search similar to the process of determining the shortest path to a specific vertex by a binary search over the space of rays in the bundle.

A similar analysis is true for the bundles that arise due to a critical source on edge \((a, b)\). We next determine how to efficiently determine the splitting of sibling pairs that define a bundle.

5.4 Splitting a sibling pair

When a sibling pair needs to be split i.e., when a ray bundle needs to be partitioned into two, based on the origin of the two rays in the sibling pair being considered, the following procedures are invoked.

5.4.1 Pair that originates from a critical segment

Consider a sibling pair \( r, r' \) that originated from a critical segment \( \kappa \). See Fig. 8. Let \( \mathcal{E} \) be the edge sequence of \( r \) and \( r' \) till they incident to an edge \( e \) of face \( f \) and let \( B \) be the corresponding
ray bundle. Let $r$ be incident to $x \in e$ and let $r'$ be incident to $x' \in e$. Given that $r$ and $r'$ are parallel between any two successive edges in $E$, following are the possibilities: $r$ and $r'$ refract from $x$ and $x'$ respectively at the same angle; $r$ and $r'$ critically reflect from $x$ and $x'$. Since the latter kind of rays are not geodesic paths (Proposition \[\square\]), we focus on the former. If the rays refracted from $x$ and $x'$ both incident to the same edge of $f$, then there is no need to split the sibling pair $r, r'$. Otherwise, as explained below, we find new rays $r'_1$ and $r'_2$ originated from $\kappa$ to respectively pair with $r$ and $r'$.

Let $v_1, v_2, v_3$ be the vertices of $f$ where the end points of $e$ are $v_1, v_2$. Let $l$ be a line segment on face $f$ such that it is parallel to line segment $f \cap r$ ($r$ refracted at $p$) and passes through vertex $v_3$ of face $f$ and its other endpoint is $x''$ on $v_1 v_2$. We interpolate over the rays in $R(\kappa)$ to find a ray $r'_1 \in R(\kappa)$ (resp. $r'_2 \in R(\kappa)$) such that the section of $r'_1$ (resp. $r'_2$) on $f$ lies between $l$ and the section of $r$ (resp. $r'$) on $f$. Then the sibling pair $r, r'$ splits into: $r, r'_1$ and $r', r'_2$. Since between any successive edges in $E$, rays $r$ and $r'$ are parallel, the interpolation in possible. Based on the ratio of $xx''$ to $xx'$, we interpolate a point $q$ on $\kappa$ so that a critically reflected ray $r''$ from $q$ reaches $v_3$. The ray $r'_1$ (resp. $r'_2$) is the one whose origin is closest to $q$ among all the rays between $r'''$ and $r$ (resp. $r'$). These sibling pairs are pushed to the event heap with the key values being the respective weighted Euclidean distances' from $s$ to the points at which these pairs strike the edges $v_1 v_3$ and $v_2 v_3$.

Let $r'_1$ incident to $v_1 v_3$ at $z'_1$ and let $r'_2$ incident to $v_2 v_3$ at $z'_2$. Further, an event point to initiate a discrete wavefront from $v_3$ is pushed to the heap with the key value $\min(d_{z'_1} + w_{v_1 v_3}\|z'_1 v_3\|, d_{z'_2} + w_{v_2 v_3}\|z'_2 v_3\|)$ where $d_{z'_1}$ is the weighted Euclidean distance from $s$ to $z'_1$ and $d_{z'_2}$ is the weighted Euclidean distance from $s$ to $z'_2$.

### 5.4.2 Pair that originates from a tree of rays

Let $e_i, e_j, e_k$ be the edges bounding face $f$. Also, let $r', r''$ be a sibling pair of $T_R(w)$, till both of them strike edge $e_i$ of $f$. With further expansion of the wavefront, let $r'$ strike edge $e_j$ at $q'$ and let $r''$ strike edge $e_k$ at $q''$. This requires us to split the sibling pair $r', r''$. Let $p'$ (resp. $p''$) be the critical ancestor path of $r'$ (resp. $r''$). Further, let $REG$ be the open region bounded by $p', p'', r', r''$ and the line segment $q'q''$. Let $R_1$ (resp. $R_2$) be the set of rays such that for a ray $r \in R_1$ (resp. $r \in R_2$) if and only if $r$ originates from a critical point of entry or vertex located on the critical ancestor path $p'$ (resp. $p''$) and the ray $r$ lies in $REG$. With binary search over the rays in $R_1$ we find a ray $r_1 \in R_1$; similarly, with binary search over $R_2$ we find a ray $r_2$ such that $r_1$ intersects $e_j, r_2$ intersects $e_k$, and $r_1$ and $r_2$ are either successive rays originating from the same origin or adjacent origins on a critical ancestor path. This lets us splitting a ray bundle of $T_R(w)$ into two: with $r_1, r'$ and $r'', r_2$ sibling pairs of $T_R(w)$. As explained in previous subsections, we push appropriate event points to the event queue reflecting that these are the siblings. Further, we also push the event corresponding to initiating a discrete wavefront from $v$. Note that splitting a sibling pair originated from a vertex is just a specialization of the procedure listed above.

The following theorem shows that while searching for a ray located in a cone of rays and when that cone angle is small, instead of tracing a ray across $E$, it suffice to interpolate the position and the angle of refraction of $r$ from the last edge of $E$.

**Theorem 5.1** Let $r', r''$ be two siblings of a ray bundle $B$, with angle between them and between the rays and the edges of the region $P$ at least $\theta$, where $\theta$ is upper bounded by $\frac{\pi}{2^3}$. Let $E$ be the edge sequence associated with $B$ and let $e$ be the last edge in $E$. Further, when a ray $r \in B$ is traced across $E$, let $q \in e$ be its point of incidence on $e$ and let $\theta'$ be the angle of refraction of $r$ from $e$. Both the position $q$ and the angle of refraction $\theta'$ can be approximated with an error $(1 + \frac{\pi}{2^3})$.

**Proof:** Let the edges in $E$, angles of incidence, and the angles of refractions of rays $r'$ and $r''$ across $E$ are as mentioned in Lemma \[\square\]. Note that the angles of refractions of $r'$ and $r''$ from $e$ are $\theta_p$ and $\theta_p + \delta_p$ respectively. Assuming that $\theta_j, \forall j$ is bounded away from 90 degree, we derive a bound on the maximum dispersion of the rays

$$
\delta_p = \frac{w_p}{w_p} 3 \Pi_{i} \frac{\cos(\theta_i + \delta_i)}{\cos(\theta_{i+1} + \delta_{i+1})} \\
\leq \frac{w_p}{w_p} 3 \Pi_{i} \frac{\cos(\theta_i') - \delta'_i \sin(\theta_i')}{\cos(\theta_{i+1}' - \delta_{i+1} \sin(\theta_{i+1}'))}
$$
Therefore, to keep the error ratio \( \frac{\delta'_1}{\delta'_1 - \gamma_p} \) less than \( \epsilon \), we choose \( \delta'_1 \leq \frac{\epsilon}{n^{\gamma_1}} \) since \( \cos \theta_i \geq 1/n^\gamma_1 \).

In fact we choose \( \delta'_1 \leq \frac{\epsilon}{n^{\gamma_1}} \) so as to keep the error ratio limited to \( \epsilon/n^2 \) and hence bound the error in the distance as outlined next.

Let \( q_1, q_2 \ldots \) be the points along the edges where \( r \) strikes the edge sequence \( \mathcal{E} \). The path traversed by the ray \( r \) is a piece-wise linear sequence of lengths, say \( d_1, d_2 \ldots d_j \). The error induced due to the error in choosing the interpolated ray \( \hat{r} \) is \( \sum_i d_i \epsilon/n^2 \) giving a total error of a factor of \( \epsilon \) in the length of the path as well as in the position of \( \hat{r} \) on the last edge \( e \).

\[ \mathcal{E} \]

From Theorem 4.1 and Theorem 4.2 we know that for set \( S \) of rays originated from a vertex, or a critical source, or a tree of rays, the cardinality of \( S \) is \( \Theta\left(\frac{4\mu}{\epsilon^2} \left(\frac{K}{n^2}\right)^{n^2}\right) \). Let \( \mathcal{E} \) be an edge sequence of a sibling pair \( r', r'' \), and let \( e \) be the last edge of \( \mathcal{E} \). In doing binary search over \( S \) to find a ray \( r \in S \), we need to trace \( O(\lg(\frac{4\mu}{\epsilon^2} + n^2 \lg \frac{K}{n^2}) \) rays in \( S \) to \( e \). Considering the cardinality of \( \mathcal{E} \) (which is \( O(n^2) \)) from the Proposition 2, the time it takes to find a ray from the source is \( O(n^2 \lg(\frac{4\mu}{\epsilon^2} + n^2 \lg \frac{K}{n^2})) \). Herewith, we devise an algorithm to improve upon this efficiency.

We partition the \( O(\frac{4\mu}{\epsilon^2} \left(\frac{K}{n^2}\right)^{n^2}) \) rays in \( S \) into a hierarchy: zeroth level has the entire ordered set of rays (ordered with respect to the angle made with \( x \)-axis) while \( i^{th} \)-level has \( O(\frac{4\mu}{\epsilon^2} \left(\frac{K}{n^2}\right)^{ni}) \) rays. (There are \( O(n) \) levels in this tree.) An ordered set \( S' \) of rays associated with a node is partitioned into \( O\left(\frac{4\mu}{\epsilon^2} \left(\frac{K}{n^2}\right)^{ni}\right) \) ordered sets at its children such that the cone angle associated with \( S' \) is partitioned at its children i.e., all the rays in a cone are associated with a child. Observe that the angle range covered by any node decreases with the depth of the tree. In the binary search corresponding to the first \( k \)-levels, we do trace the rays of interest across \( \mathcal{E} \). By exploiting
Theorem 5.1 we interpolate the point of incidence of \( r \) together with the angle of refraction from \( e \) in the \( k + 1^{st} \) level. Further, to satisfy Theorem 5.1 we choose \( k = \frac{\ln n}{\lg \frac{1}{\epsilon}} \) so that the angle between two successive rays in level \( k \) is less than or equal to \( \frac{\epsilon}{n^{1/2}} \).

### 5.5 Determining Shortest Path to an Edge via a Bundle

Consider a bundle \( B \) and the edge sequence that \( B \) traverses, termed \( E \) where the last edge is, say, \( e \). To process the events corresponding to the strike of bundles we determine the shortest paths from the vertex that initiated the bundle \( B \) to the edge \( e \). By using a process similar to that in the previous section the shortest path to the portion of the edge in-between the points struck by the sibling rays of \( B \), termed \( e_B \), can be determined. W.l.o.g. we can assume that all the rays in \( B \) strike all edges in the edge sequence \( E \) at an incidence angle in between 0 and 90 and thus do not span the normal. The shortest distance to the edge \( e \) will be determined by a binary search over the rays, given the monotone nature of the shortest distance to points on edge \( e \).

### 6 Analysis

**Lemma 6.1** The algorithm correctly determines sibling rays in \( T_B(u) \) for every source \( u \), and computes an \( \epsilon \)-approximation to a Type-1 shortest path from \( u \) to \( v \) for any two sources \( u, v \).

**Proof:** An optimal path \( P \) can be partitioned into sub-paths, each sub-path going from one vertex \( v \) to another vertex \( w \). Suppose a sub-path uses a critical segment \( \kappa \) and is partitioned as follows: a path \( P_1 \) from \( v \) to \( \kappa \) and a path \( P_2 \) from \( \kappa \) to \( w \). Let \( e \) be the edge on a face \( f \) such that the critical segment \( \kappa \) lies on \( e \) and reflects rays back onto face \( f \). Also, let \( y \) be the critical point of entry into \( \kappa \). We first show that a good approximation to the path from \( \kappa \) to \( w \) can be found. If \( w \) is the endpoint of \( \kappa \) then we are done since the distance to the endpoint from the critical source is included in consideration. Otherwise consider rays generated from \( \kappa \) that are parallel and separated by a weighted Euclidean distance of \( \delta \). Let \( w \) be a vertex between a sibling pair \( r_1, r_2 \) originated from \( \kappa \). If we choose \( \delta \leq \epsilon' \), then as in Lemma 4.2 the error in computing the distance from \( w \) to a point on the critical segment is bounded by \( d \mu (1 + \frac{1}{\sin \delta_{\min}}) \epsilon' \), where \( d \) is the weighted Euclidean distance of \( w \) from \( \kappa \). The optimal path \( P_2 \) has a first segment that is parallel to and lies in between a pair of rays \( r' \) and \( r'' \). The source of the optimum path lies in between the origin of the two rays, \( \text{origin}(r') \) and \( \text{origin}(r'') \), that lie on \( \kappa \). Since the weighted Euclidean distance between \( \text{origin}(r') \) and \( \text{origin}(r'') \) is at most \( \delta \), the error in determining a path along one of these rays is \( \delta + d \mu (1 + \frac{1}{\sin \delta_{\min}}) \epsilon' \). Further the error in determining the shortest path from \( v \) to \( y \) is at most \( O(d_1 \epsilon' n^2 \mu (1 + \frac{1}{\sin \theta_{\min}})) \) where \( d_1 \) is the optimal distance from \( v \) to \( \kappa \). Combining the above errors, and noting that there are at most \( O(n) \) vertices on the shortest path gives the required error bound as in Theorem 4.1.

**Lemma 6.2** Let \( P \) be a Type-2 shortest path from a vertex \( v \) to another vertex \( w \) on \( \mathcal{P} \) with a critical segment \( \kappa \) in-between. Then the algorithm determines a pair of traced rays in \( \mathcal{R}(v) \) that can approximate the shortest path, \( P \).

**Proof:** Suppose that there exists a Type-1 shortest path between \( u \) and \( v \), traversing a sequence of edges \( \mathcal{E} \). Lemma 4.1 shows that this path can be found using sibling rays. We show that the algorithm maintains sibling rays for edges in \( \mathcal{E} \) that enables the determinations of traced rays such that these rays can be used to find an \( \epsilon \)-approximate shortest path from \( u \) to \( v \). First consider the case in which a shortest path lies in between two successive rays each of which intersect the sequence \( \mathcal{E} \). These two successive rays will be found from sibling pairs by the algorithm. The second case occurs when the shortest path uses a vertex \( v \). In the second case, the vertex \( v \) must be less than \( d_u(v) \epsilon' \) from a ray (in fact a traced sibling ray) in \( \mathcal{R} \), since two points at distance \( d \) on two successive rays, are separated by a distance of at most \( de' \) (Lemma 4.1). The algorithm
computes the estimate of the shortest distance to a vertex \( v \) whenever sibling rays are computed at every edge incident to \( v \).

It thus suffices to show that sibling rays are computed correctly. Initially, the siblings are computed correctly and each ray is refracted correctly. Consider the procedure that computes a sibling pair \( r_1, r_2 \) closest to a vertex \( v \). The rays \( r_1 \) and \( r_2 \) are thus two successive rays and the distance to \( v \) via the rays \( r_1 \) and \( r_2 \) is updated. Suppose both the rays \( r_1 \) and \( r_2 \) have the same origin. We find the siblings and due to Lemma 4.1, an \( \epsilon \)-approximation to the shortest distance from origin of \( r' \) to \( v \) is ensured. Otherwise, \( \text{origin}(r_1) \neq \text{origin}(r_2) \). In this case, we find a sibling pair via the binary search on the rays in the critical ancestor paths of two siblings, say \( r' \) and \( r'' \) such that the rays \( r_1 \) and \( r_2 \) are part of the set of rays lying in between \( r' \) and \( r'' \). Let \( r \) be the ray which is initiated at the origin of \( r_2 \). And let ray \( r \) has as its origin, \( p \), which is same as the origin of \( r_1 \). Also, we know that the ray \( r_1 \) is a successor to ray \( r \) since it is adjacent to \( r_2 \). Let the origin of \( r_2 \) be located on an edge \( e' \). Suppose \( r_1 \) intersects \( e' \) at point \( p_1 \). As described, by construction, ray \( r_1 \) is parallel to \( r_2 \), starting at \( p_1 \). Due to Lemma 4.1, the distance between \( p_1 \) and the origin of \( r_2 \) is bounded. These facts, together with Theorems 4.2 and 4.1, guarantee the \( \epsilon \)-approximation in computing the shortest distance from \( \text{origin}(r') \) to \( v \). \( \square \)

**Theorem 6.1** The algorithm computes an \( \epsilon \)-approximate shortest path from \( s \) to \( t \) correctly.

**Proof:** Provided the bundles are maintained correctly the theorem follows immediate from the last two lemmas. That the bundles are maintained correctly, follows from the fact that bundles are eliminated only when it is determined that they are crossed over by another bundle and hence not necessary for shortest path computation. \( \square \)

**Theorem 6.2** The algorithm computes an \( \epsilon \)-approximate shortest path from \( s \) to \( t \) in \( O(n^4(n \log n + \log(\frac{n}{\epsilon} (1 + \frac{1}{\sin \theta_{\text{min}}}))))) \) time complexity.

**Proof:** As mentioned in Subsection 5.1 to improve the time complexity, the algorithm exploits Proposition 4. We need to show that the number of ray bundles at any instant of the algorithm are \( O(n) \). The number of bundles that are initiated from vertex sources are \( O(n) \).

Furthermore, by Proposition 3 the number of bundles that are generated from critical segments is bounded by \( O(n^k) \). The processing of such bundles will be handled separately.

Bundles increase when split, however that increased is charged to the vertex that causes the split. Note that during the split, multiple bundles are split but the propagation of the bundles from an edge is via the shortest path distance to the edge. If it is determined during the event handling via the heap that the shortest distance to the edge has been reached then the bundle is useful and is propagated further across the adjacent face.

Determining the ray originating from a critical segment and that strikes a vertex can be determined via a binary search on the space of the parallel rays that are part of the bundle that originates from a critical segment. The work involved in tracing a ray that originates at a critical segment and traverses an edge sequence takes \( O(n^2) \) time (Proposition 5). The specific pair of successive rays of interest are found by interpolation, taking \( O(1) \) time. As there are \( O(n) \) vertices and \( O(n^2) \) critical segments (Proposition 3), the time complexity is \( O(n^5) \).

We thus need to analyze the time involved in splitting the ray bundles. Let \( S \) be the set of rays from a vertex source \( v_1 \). While searching for a ray in \( S \) so that \( r \) strikes the last edge \( e \) of an edge sequence \( \mathcal{E} \), by imposing a hierarchy over \( S \) (as described above) we do binary search in the first \( k \) levels of the hierarchy and interpolate (instead of trace) in the \( k + 1 \)st level. Hence, the work done at a vertex \( v \) while the rays from \( v_1 \) are considered is equal to \( O(n^2 \log \frac{n}{\epsilon} + n^3k \log \frac{1}{\epsilon}) \). Assuming that \( j \) vertices were traversed by the discrete wavefront, in the worst-case we may need to split \( j \) bundles at a vertex \( v \) which is not yet struck by the wavefront. Hence, the total work in traversing all the vertices is \( O(n^4 \log \frac{n}{\epsilon} + n^3k \log \frac{1}{\epsilon}) \). As \( k \) is chosen to be equal to \( \frac{\log n}{\log \frac{1}{\epsilon}} \), this becomes \( O(n^4 \log \frac{n}{\epsilon} + n^5 \log n) \). The same complexity is achieved when the ray bundle splits are considered at critical points of entry.
Let us consider the last case in which the sibling pair is from a tree of rays, say $T_R(u)$. Let $S$ be the set consisting of all the critical points of entry in any critical ancestor path $P$ in $T_R(u)$. If we do a binary search over the set of rays in $R(v')$, for some $v' \in S$, we need not do binary search over $R(v'')$ for some $v'' \in S$ where $v'' \neq v'$. It then remains to determine the vertex $v' \in S$ where the binary search is required. Since $S$ is $O(n^2)$ in size and since it is ensured that no two ray bundles cross (Section 5.1), the vertex of interest can be found in $O(n^2)$ time. Hence, the time complexity remains same as in the previous case.

7 Conclusions

In this paper we present a polynomial-time algorithm for finding an approximate weighted Euclidean shortest path between two given points $s$ and $t$. The main ideas of this algorithm rely on progressing the discretized wavefront from $s$ to $t$. The time complexity of our algorithm is $O(n^2 \text{ poly-log})$; significantly, the algorithm is polynomial with respect to the parameters. This result is about a cubic factor (in $n$) improvement over the Mitchell and Papadimitriou’s ‘91 result [8], which is the only known polynomial time algorithm for this problem to date. Since the lower bound on the problem (from [8]) stands at $\Omega(n^4)$, it would be interesting to explore further the time complexity in devising an approximation scheme.

References

[1] L. Aleksandrov, H. Djidjev, H. Guo, A. Maheshwari, D. Nussbaum, and J.-R. Sack. Algorithms for approximate shortest path queries on weighted polyhedral surfaces. Discrete & Computational Geometry, 44(4):762–801, 2010.

[2] L. Aleksandrov, M. Lanthier, A. Maheshwari, and J.-R. Sack. An $\epsilon$-approximation for weighted shortest paths on polyhedral surfaces. In 6th Scandinavian Workshop on Algorithm Theory, Stockholm, Sweden, pages 11–22, 1998.

[3] L. Aleksandrov, A. Maheshwari, and J.-R. Sack. Determining approximate shortest paths on weighted polyhedral surfaces. Journal of the ACM, 52(1):25–53, 2005.

[4] S.-W. Cheng, H.-S. Na, A. Vigneron, and Y. Wang. Querying approximate shortest paths in anisotropic regions. SIAM Journal on Computing, 39(5):1888–1918, 2010.

[5] M. Lanthier, A. Maheshwari, and J.-R. Sack. Approximating shortest paths on weighted polyhedral surfaces. Algorithmica, 30(4):527–562, 2001.

[6] C. Mata and J. Mitchell. A new algorithm for computing shortest paths in weighted planar subdivisions (extended abstract). In 13th Symposium on Computational Geometry, pages 264–273, 1997.

[7] J. Mitchell. Geometric shortest paths and network optimization. In J.-R. Sack and J. Urrutia, editors, Handbook of Computational Geometry, pages 633–701. Elsevier, 2000.

[8] J. Mitchell and C. Papadimitriou. The weighted region problem: Finding shortest paths through a weighted planar subdivision. Journal of the ACM, 38(1):18–73, 1991.

[9] Z. Sun and J. Reif. On finding approximate optimal paths in weighted regions. Journal of Algorithms, 58(1):1–32, 2006.