Approximate shortest paths amid weighted regions

R. Inkulu
http://www.iitg.ac.in/rinkulu/
Associate Professor
Department of CSE
IIT Guwahati, India

(Joint work with Prof. Sanjiv Kapoor)
Outline

1 Introduction

2 Literature

3 Our algorithm

4 Conclusions
Motivation

Modeling region specific constraints in motion planning:

- walking on bushy region vs on plain road
- terrain navigation of an autonomous vehicle
Problem description

The cost of the red path is

\[ w_{f_1} \|sa\| + w_{f_2} \|ab\| + w_{f_3} \|bc\| + \min(w_{f_3}, w_{f_4}) \|cd\| + w_{f_3} \|dg\| + w_{f_5} \|gt\| \]

- Given a triangulation \( \mathcal{P} \) with \( O(n) \) faces, each face associated with a positive weight, find a path between two input points \( s \) and \( t \) (both belonging to \( \mathcal{P} \)) so that the path has minimum cost among all possible paths joining \( s \) and \( t \) that lie on \( \mathcal{P} \).

- 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 lies.
Outline

1. Introduction
2. Literature
3. Our algorithm
4. Conclusions

(Approximate shortest paths amid weighted re...
[Mitchell, Papadimitriou JACM ’91]: main contributions

- Characterized weighted shortest path in terms of Snell’s laws of light

- Constructed a shortest path map with respect to a point $s$ using continuous Dijkstra wavefront

- Lower bounded the number of events involved in the continuous Dijkstra amid weighted regions as $\Omega(n^4)$
If a geodesic path $p$ shares a segment $yy'$ with edge $e$, for $y$ not being a vertex, then both the angle of incidence at $y$ from $f'$ and angle of exit into $f'$ at $y'$ are critical: $\theta_c = \sin^{-1}\left(\frac{w_{f'}}{w_{f''}}\right)$. (This case arise only when $w_{f'} < w_{f''}$.)

(Assume shortest paths amid weighted regions)
[Mitchell, Papadimitriou JACM ’91]: characterizing shortest paths in terms of Snell’s laws

• If a geodesic path \( p \) shares a segment \( yy' \) with edge \( e \), for \( y \) not being a vertex, then both the angle of incidence at \( y \) from \( f' \) and angle of exit into \( f' \) at \( y' \) are critical: \( \theta_c = \sin^{-1}\left(\frac{w_{f'}}{w_{f''}}\right) \). (This case arises only when \( w_{f''} > w_{f'} \).)

• If a geodesic path \( p \) crosses edge \( e \) at a point \( z \), then \( p \) obeys Snell’s law of refraction at \( z \): \( w_{f'} \sin\theta' = w_{f''} \sin\theta'' \) for \( \theta' < \theta_c \).
Simulates the wavefront progress by the progression of intervals of optimality over the faces of $\mathcal{P}$.
[Mata, Mitchell SoCG ’97]: both in geometric space and as a graph-theoretic problem

Constructs a spanner $G(V, E)$ wherein

- $V$ comprises of all the vertices and the critical points of entry

- For each cone $C$ in $k$ evenly-spaced cones $C$ placed around each $v \in V$, find a split vertex or a critical point of entry $u$ that belong to $C$ and join $v$ and $u$ with an edge
[Lanthier et al. Algorithmica ’01]: reduced to a graph-theoretic problem

For each face $f_i$ of $\mathcal{P}$ a graph $G_i$ is constructed: $\Theta(n^2)$ Steiner points are evenly placed along each edge of $f_i$; a node pair $u$ and $v$ is connected in $G_i$ whenever $u$ and $v$ belong to distinct edges of $f_i$ or they are neighbors on an edge.

outputs an apprx shortest path with additive error
Variants of [Lanthier et al. Algorithmica ’01]

- [Aleksandrov et al. SWAT’98]
  Steiner points are placed in a geometric progression along the edge

- [Aleksandrov et al. STOC’00]
  Based on Snell’s laws of refraction, prunes edges through which Dijkstra’s wavefront need to progress

- [Sun, Reif JAlgo ’06]
  Prunes further by exploiting the non-crossing property of shortest paths

- [Aleksandrov et al. JACM ’05]
  Steiner points are placed in a geometric progression along the three bisectors of each face

- [Cheng et al. SIAMJC ’10]
  First prunes $\mathcal{P}$ based on the intersection of an ellipse (whose size relies on the unweighted geodesic distance between $s$ and $t$) and $\mathcal{P}$ before applying [Aleksandrov et al. JACM ’05]

(Approximate shortest paths amid weighted re...
## Time complexity comparison chart

<table>
<thead>
<tr>
<th>Reference</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Mitchell, Papadimitriou JACM '91]</td>
<td>$O(n^8 \lg \frac{nN\mu}{\epsilon})$</td>
</tr>
<tr>
<td>[Mata, Mitchell SoCG '97]</td>
<td>$O(\frac{\mu}{\epsilon \theta_{\text{min}}} n^3)$</td>
</tr>
<tr>
<td>[Sun, Reif JAlgo '06]</td>
<td>$O(\frac{nN^2}{\epsilon} \lg(N\mu) \lg \frac{n}{\epsilon} \lg \frac{1}{\epsilon})$</td>
</tr>
<tr>
<td>[Aleksandrov et al. JACM '05]</td>
<td>$O(\frac{nN^2}{\sqrt{\epsilon}} \lg(N\mu) \lg \frac{n}{\epsilon} \lg \frac{1}{\epsilon})$</td>
</tr>
<tr>
<td>[Cheng et al. SIAMJC '10]</td>
<td>$O(\frac{\rho^2 n^3}{\epsilon^2} (\lg \frac{\rho n}{\epsilon})^2)$</td>
</tr>
<tr>
<td>Our submitted result</td>
<td>$O(n^5 \lg n + n^4 \lg(\frac{\mu}{\epsilon} (1 + \frac{1}{\sin \theta_{\text{min}}})))$</td>
</tr>
</tbody>
</table>

Like [Mitchell, Papadimitriou JACM '91], our algorithm is polynomial in $n$.

---

1 $n$: 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 $w_{\text{min}}$; $k$ is the smallest integer such that the sum of the $k$ smallest angles in $P$ is at least $\pi$; $\rho$: $w_{\text{max}}$ excluding $\infty$. 

(Approximate shortest paths amid weighted re
Outline

1. Introduction
2. Literature
3. Our algorithm
4. Conclusions
Progressing wavefront: continuous Dijkstra

assuming weights of all the faces are same
Progressing wavefront: discretized Dijkstra

(Approximate shortest paths amid weighted regions)
Tracing discrete wavefront

successive rays

sibling rays

sibling rays of a ray bundle are in the same color

initiate many rays from $s$ but trace only few

(Approximate shortest paths amid weighted re)
Events corresponding to tracing of ray bundles

binary search in determining the event points of interest

following events are pushed to min-heap:

  event point pair: \( q'_1 - q''_1 \), etc.,

  events point pair: \( q'_1 - q''_2 \), etc.,
Initiating ray bundles from a vertex

initiate a discrete wavefront from $v$ when blue ray bundle strikes $v$ while exploiting the non-crossing property of shortest paths
two rays belong to a ray bundle if they traverse across the same edge sequence whenever traced
Ray bundle split due to a vertex

successive rays $r'_1$ and $r''_1$ are identified with binary search over the rays in blue ray bundle

- The rays that belong to the same ray bundle, the edge sequence that they traverse across is same.
Ray bundle split due to a vertex

new ray bundles are formed and the corresponding sibling pairs are defined

(Approximate shortest paths amid weighted re
Detecting critical incidence

here the critical angle $\theta_c$ is $\sin^{-1}\left(\frac{w_{f'}}{w_{f''}}\right)$ wherein $w_{f'} < w_{f''}$
Initiating rays from a critical segment

(Approximate shortest paths amid weighted re)
Tracing rays from a critical segment

(Approximate shortest paths amid weighted re)
Split of a ray bundle initiated at a critical segment

- Linear interpolation in finding $x''$ suffice instead of tracing rays from $\kappa$
Rays initiated from a critical source

- Helps in having sparser sets of rays initiated from vertex and critical segment sources
- These rays are traced similar to the way rays initiated from a vertex source

(Assuming approximate shortest paths amid weighted regions)
Recap

sources of ray bundles:
Recap

sources of ray bundles:
  - vertices of \( \mathcal{P} \), including \( s \)
Recap

sources of ray bundles:

- vertices of $\mathcal{P}$, including $s$
- critical segments
Recap

sources of ray bundles:

- vertices of $\mathcal{P}$, including $s$
- critical segments
- critical sources
Recap

sources of ray bundles:
  • vertices of $\mathcal{P}$, including $s$
  • critical segments
  • critical sources

event points of interest:
Recap

sources of ray bundles:

- vertices of $\mathcal{P}$, including $s$
- critical segments
- critical sources

event points of interest:

- initiating rays from ray bundle sources
Recap

sources of ray bundles:
- vertices of $\mathcal{P}$, including $s$
- critical segments
- critical sources

event points of interest:
- initiating rays from ray bundle sources
- tracing ray bundles
Recap

sources of ray bundles:

• vertices of $\mathcal{P}$, including $s$
• critical segments
• critical sources

event points of interest:

• initiating rays from ray bundle sources
• tracing ray bundles
• ray bundle splits due to new ray bundle sources
Algorithm

(1) initiate a set of ray bundles from s
Algorithm

(1) initiate a set of ray bundles from $s$

(2) while ($t$ is not struck by a ray bundle)
Algorithm

(1) initiate a set of ray bundles from $s$

(2) while ($t$ is not struck by a ray bundle)
   
   (i) push new event points to min-heap
Algorithm

(1) initiate a set of ray bundles from $s$

(2) while ($t$ is not struck by a ray bundle)
   (i) push new event points to min-heap
   (ii) handle event points popped from min-heap
Few optimizations: tree of rays

- ray bundles from each vertex source $v$, and
Few optimizations: tree of rays

- ray bundles from each vertex source $v$, and
- ray bundles from critical sources whose nearest ancestor vertex is $v$ are organized into a tree, $T_R(v)$

(Approximate shortest paths amid weighted re...
Few optimizations: tree of rays

- ray bundles from each vertex source $v$, and

- ray bundles from critical sources whose nearest ancestor vertex is $v$ are organized into a tree, $\mathcal{T}_R(v)$

For each $v$, we do binary searches across ray bundles in $T_R(v)$. 
Few more optimizations: interpolate when the angle is small

- avoid tracing rays across lengthy ($O(n^2)$) edge sequence: instead interpolate when the angle between traced rays is small
Properties exploited in the analysis

• 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 $p$ can be modified in such a way that case (i) holds without altering the length of the path.
Properties exploited in the analysis

- 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 $p$ can be modified in such a way that case (i) holds without altering the length of the path.

- The length of any edge sequence of a shortest locally $f$-free path $p$ is $O(n^2)$. 

(Approximate shortest paths amid weighted re
Properties exploited in the analysis

• 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 $p$ can be modified in such a way that case (i) holds without altering the length of the path.

• The length of any edge sequence of a shortest locally $f$-free path $p$ is $O(n^2)$.

• Any shortest geodesic path $p$, passes through $O(n)$ critical points of entry on any given edge $e$. 
Properties exploited in the analysis

• 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 $p$ can be modified in such a way that case (i) holds without altering the length of the path.

• The length of any edge sequence of a shortest locally $f$-free path $p$ is $O(n^2)$.

• Any shortest geodesic path $p$, passes through $O(n)$ critical points of entry on any given edge $e$.

• Non-crossing property of shortest paths: Any two shortest geodesic paths with the same source point cannot intersect in the interior of any region.
Bounding the ray density at sources to obtain a PTAS

Considering refraction/reflection paths of any two successive rays initiated from any type of source, initiating \( O\left(\frac{2\mu}{\epsilon'} \left(\frac{1}{\epsilon'}\right)^2 n^2\right) \) suffice to achieve \( \epsilon \)-approximation, where \( \epsilon' = \frac{\epsilon}{n^3 \mu (1 + \frac{1}{\sin \theta_{\min}})} \).
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources
- splits of ray bundles from critical segments
- tracing ray bundles across edge sequences
- savings due to interpolations

\[ T \in O\left(n^5 \log n + n^4 \log \left(\frac{\mu}{\epsilon} \left(1 + \sin \theta_{\min}\right)\right)\right) \]

time to find an \( \epsilon \)-approximate shortest path from \( s \) to \( t \).

(Approximate shortest paths amid weighted regions)
Time complexity

- ray bundle splits at vertices

\[
O\left(n^5 \log n + n^4 \log \left(\mu \epsilon \left(1 + \frac{1}{\sin \theta_{\min}}\right)\right)\right)
\]
time to find an \(\epsilon\)-approximate shortest path from \(s\) to \(t\).

(Approximate shortest paths amid weighted re...
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources

\[ T \approx O(n^5 \log n + n^4 \log (\mu \epsilon (1 + 1 / \sin \theta_{\text{min}}))) \]
time to find an \( \epsilon \)-approximate shortest path from \( s \) to \( t \).

(Approximate shortest paths amid weighted re...
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources

\[ T = O\left(n^5 \log n + n^4 \log \left(\frac{\mu}{\epsilon} \left(1 + \frac{1}{\sin \theta_{\min}}\right)\right)\right) \]

time to find an \( \epsilon \)-approximate shortest path from \( s \) to \( t \).
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources
- splits of ray bundles from critical segments

\[\text{Takes } O\left(n^{5.5 \log n} + n^4 \log \left(\frac{\mu}{\epsilon} \right) \right) \text{ time to find an } \epsilon\text{-approximate shortest path from } s \text{ to } t.\]
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources
- splits of ray bundles from critical segments
- tracing ray bundles across edge sequences

\[ T(n) = O(n^5 \log n + n^4 \log(\mu \epsilon (1 + \frac{1}{\sin \theta_{\min}}))) \]

Takes \( O(n^5 \log n + n^4 \log(\mu \epsilon (1 + \frac{1}{\sin \theta_{\min}}))) \) time to find an \( \epsilon \)-approximate shortest path from \( s \) to \( t \).
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources
- splits of ray bundles from critical segments
- tracing ray bundles across edge sequences
- savings due to interpolations

Takes $O(n^5 \log n + n^4 \log (\mu \epsilon (1 + \frac{1}{\sin \theta_{\min}})))$ time to find an $\epsilon$-approximate shortest path from $s$ to $t$. 

(Approximate shortest paths amid weighted re
Time complexity

- ray bundle splits at vertices
- ray bundle splits at critical sources
- number of ray bundles from Steiner sources
- splits of ray bundles from critical segments
- tracing ray bundles across edge sequences
- savings due to interpolations

Takes $O(n^5 \log n + n^4 \log \left( \frac{\mu}{\epsilon} \left(1 + \frac{1}{\sin \theta_{\text{min}}} \right) \right))$ time to find an $\epsilon$-approximate shortest path from $s$ to $t$. 
Major ideas

- Discrete wavefront as sets of rays
- Partitioning the wavefront into ray bundles
- Lazy tracing of rays in ray bundles
- Viewing all the rays due to a vertex source as a tree of rays
- Interpolating instead of tracing wherever it is possible
Major ideas

- discrete wavefront as sets of rays
Major ideas

- discrete wavefront as sets of rays
- partitioning the wavefront into ray bundles
Major ideas

- discrete wavefront as sets of rays
- partitioning the wavefront into ray bundles
- lazy tracing of rays in ray bundles
Major ideas

- discrete wavefront as sets of rays
- partitioning the wavefront into ray bundles
- lazy tracing of rays in ray bundles
- viewing all the rays due to a vertex source as tree of rays

(Approximate shortest paths amid weighted re...
Major ideas

• discrete wavefront as sets of rays

• partitioning the wavefront into ray bundles

• lazy tracing of rays in ray bundles

• viewing all the rays due to a vertex source as tree of rays

• interpolating instead of tracing wherever it is possible
Constructs a spanner $G(V, E)$ while progressing discrete wavefronts from vertices in lock-step fashion:
Two-point apprx shortest path queries: preprocessing

Constructs a spanner $G(V, E)$ while progressing discrete wavefronts from vertices in lock-step fashion:

- $V$ comprises of all the vertices and the critical points of entry
Constructs a spanner $G(V, E)$ while progressing discrete wavefronts from vertices in lock-step fashion:

- $V$ comprises of all the vertices and the critical points of entry
- Whenever the discrete wavefront from $v$ strikes a vertex/critical point of entry $u$, an arc $vu$ is introduced into $G$
Two-point apprx shortest path queries: querying

- Locate two query points \( s \) and \( t \) in the triangulation
Two-point apprx shortest path queries: querying

- Locate two query points $s$ and $t$ in the triangulation
- Link them into $G$ by progressing discrete wavefronts from $s$ and $t$ (like in the above algorithm)
Two-point approx shortest path queries: querying

- Locate two query points $s$ and $t$ in the triangulation
- Link them into $G$ by progressing discrete wavefronts from $s$ and $t$ (like in the above algorithm)

achieves constructing a least cost path in $O(n^4 \lg n)$ query time while the best polynomial query time with fixed $s$ stands at $O(n^7 \text{polylog})$ ([Mitchell, Papadimitriou JACM ’91])
Outline

1 Introduction

2 Literature

3 Our algorithm

4 Conclusions
Take-homes

• A generalization of well-known Euclidean shortest path problem
Take-homes

• A generalization of well-known Euclidean shortest path problem

• Continuous vs discrete-Dijkstra wavefront
Take-homes

- A generalization of well-known Euclidean shortest path problem
- Continuous vs discrete-Dijkstra wavefront
- Reducing the geometric problem to a graph-theoretic one vs solving in the geometric domain itself
Open problems

• Since the known lower bound on the number of event points is $\Omega(n^4)$ (from [Mitchell, Papadimitriou JACM ’91]), the goal is to design an algorithm with $O(n^4 \text{ polylog})$ time.
Open problems

• Since the known lower bound on the number of event points is $\Omega(n^4)$ (from [Mitchell, Papadimitriou JACM ’91]), the goal is to design an algorithm with $O(n^4 \text{ polylog})$ time

• More efficient two-point queries
Open problems

- Since the known lower bound on the number of event points is $\Omega(n^4)$ (from [Mitchell, Papadimitriou JACM ’91]), the goal is to design an algorithm with $O(n^4 \text{ polylog})$ time

- More efficient two-point queries

- Extending to compute a shortest path map with respect to $s$
Open problems

• Since the known lower bound on the number of event points is $\Omega(n^4)$ (from [Mitchell, Papadimitriou JACM ’91]), the goal is to design an algorithm with $O(n^4 \text{ polylog})$ time.

• More efficient two-point queries

• Extending to compute a shortest path map with respect to $s$

• An exact least cost path finding algorithm on polyhedral weighted surfaces
References: polynomial time algo


References

Christian Mata and Joseph Mitchell.
A new algorithm for computing shortest paths in weighted planar subdivisions (extended abstract).

Lyudmil Aleksandrov, Anil Maheshwari, and Jörg-Rüdiger Sack.
Determining approximate shortest path on weighted polyhedral surface.

Zheng Sun and John H. Reif.
On finding approximate optimal paths in weighted regions.

Querying approximate shortest paths in anisotropic regions.

(Approximate shortest paths amid weighted re
Few more references

Mark Lanthier, Anil Maheshwari, and Jörg-Rüdiger Sack.
Approximating shortest paths on weighted polyhedral surfaces.

Lyudmil Aleksandrov, Mark Lanthier, Anil Maheshwari, and Jörg-Rüdiger Sack.
An $\epsilon$-approximation for weighted shortest paths on polyhedral surfaces

Lyudmil Aleksandrov, Anil Maheshwari, and Jörg-Rüdiger Sack.
Approximation algorithms for geometric shortest path problems
Thanks!