Anna Porazilová

THE SHORTEST PATH

Abstract

This paper describes the shortest path problem, its classification and the best known algorithms. A new algorithm for the shortest path problem is introduced and its acceleration suggested.

Key words

the shortest path, geodesic, discrete geodesic, triangulation

1 Introduction

Shortest path problems are among the fundamental problems studied in computational geometry and other areas including graph algorithms, geographical information systems (GIS), network optimization and robotics. The shortest path problem has several versions.

The geodesic shortest path problem: Given two points *s* and *t* on the surface of a polyhedron, find the shortest path on the surface from *s* to *t*. The other problem is called *the Euclidean shortest path problem* and is looking for the shortest path among the obstacles in 3D space. Whereas finding the Euclidean shortest path is NP-hard, the geodesic shortest path may be found in polynomial time. This article will concentrate on the geodesic shortest path problem.

The shortest path problem can be next categorized by the distance measure used (Euclidean, weighted), purpose (*single source shortest path problem*: the shortest path between two points or *all pairs shortest path problem:* the shortest paths between one point and all triangle vertices) and computation (*exactly, approximative*). Let ε be a real number in *(0,1)*, the path is called an *(1+ε)- approximation* of the exactly shortest path between two points if its cost is at most $1+\varepsilon$ times the cost of the shortest path. In this paper the relative error R of the approximative shortest path p is the ratio between the shortest distance of the path from the final point and the length of the path: $R = \frac{d(p,t)}{\text{lenght}(p)}$.

Table 1 shows the best previous results of geodesic shortest path problems. Most of the algorithms use front propagation or some other kind of Dijkstra's-like algorithm. In 1987 Mitchell, Mount and Papadimitriou [5] introduced the Continuous Dijkstra technique, which simulates the continuous propagation of a wavefront of points equidistant from *s* across the surface, updating the wavefront at discrete events. It is obvious that the shortest path problem is an actual problem.

Surface	Approx. Ratio	Time Complexity	Reference
Convex	1	$O(n^3 \log n)$	Sharir and Schorr (1986) [7]
Non-convex	1	$O(n^2 \log n)$	Mitchell et al. (1987) [5]
Non-convex	1	$O(n^2)$	Chen and Han (1996) [1]
Non-convex	1	$O(n \log^2 n)$	Kapoor (1999) [2]
Convex	2	$O(n)$	Hershberger and Suri (1995)
Convex	1+ε	$O(n \log(1/\varepsilon)+1/\varepsilon^3)$	Agarwal et al. (1997)
Convex	1+ε	$O(n+\frac{\log n}{\varepsilon^{1,5}}+\frac{1}{\varepsilon^3})$	Har-Peled (1999)
Convex	1+ε	$O(\frac{n}{\sqrt{\varepsilon}}+\frac{1}{\varepsilon^4})$	Agarwal et al. (2002)
Convex	1+ε	$O(\frac{\sqrt{n}}{\varepsilon^{1,25}}+f(\varepsilon^{-1,25}))$	Chazelle et al. (2003)
Non-convex	1+ε	$O(n^2 \log n+\frac{n}{\varepsilon}\log\frac{1}{\varepsilon}\log\frac{n}{\varepsilon})$	Har-Peled (1999)
Non-convex	7(1+ε)	$O(n^{\frac{5}{3}} \log^{\frac{5}{3}} n)$	Varadarajan and Agarwal (2000)
Non-convex	15(1+ε)	$O(n^{\frac{8}{5}} \log^{\frac{8}{5}} n)$	Varadarajan and Agarwal (2000)

Table 1. Overview of the best algorithms for the shortest path problem

2 Computing shortest path by force of geodesic

Geodesic curves generalize the concept of straight lines for smooth surfaces and play an important role in computational geometry and GIS systems. In 2.1 a geodesic and a discrete geodesic are described. The former algorithm for geodesic computation is introduced in 2.2 and adjusted to the shortest path computation in 2.3. In 2.4 the acceleration for shortest path computation is suggested.

2.1 Geodesic curves

The well-known definition of geodesic is that a geodesic vanish the geodesic curvature. On the smooth surfaces a geodesic is the locally shortest curve.

Proposition 1 The following properties are equivalent:

1. γ is a geodesic.
2. γ is the locally shortest curve.
3. γ'' is parallel to the surface normal.
4. γ has vanishing geodesic curvature $\kappa_g = 0$

Item 2 tells that the shortest smooth curve joining two points *s* and *t* is a geodesic. The converse is not true in general. Nevertheless, the property of being shortest is desirable for curves in many applications and it is perhaps the characterization of geodesic curve more used in practice.

When trying to generalize geodesics to discrete surfaces we encounter some obstacles. It is not possible in general to find a set of curves over discrete surfaces for which all items of proposition 1 are valid.

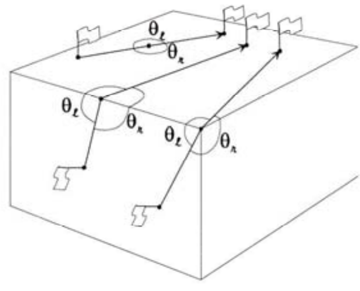


Figure 1. Right and left angles (θ_r and θ_l resp.) in a curve.

There are two different generalizations of geodesic curves to a discrete surface, both of them are called *discrete geodesics* [4]. The *shortest geodesics* are the locally shortest curves on the surface. The *straightest geodesics* satisfy the item 4 of proposition 1. The *discrete geodesic curvature* is a generalization of the geodesic curvature. Let θ be the sum of incident angles at a point *P* of a curve γ on the surface and θ_r and θ_l the respective sum of right and left angles (see figure 1), the discrete geodesic curvature is defined as

$$\kappa_g(P) = \frac{2\pi}{\Theta}\left(\frac{\Theta}{2} - \Theta_r\right).$$

Choosing θ_l instead of θ_r changes the sign of κ_g. The straightest geodesic is a curve with zero discrete geodesic curvature at each point. In other words, straightest geodesics always have $\theta_r = \theta_l$ at every point.

2.2 Geodesic computation

In 2004 in my diploma thesis [6] I implemented a geometrical algorithm for geodesic computation, which was described by Hotz and Hagen in 2000 [3]. The algorithm works on a triangulated surface, and given a start point and an initial direction computes the straightest geodesic. When encountering a vertex or an edge, the next part of discrete geodesic leads in such direction so that the left and right angles are equal (see fig. 2).

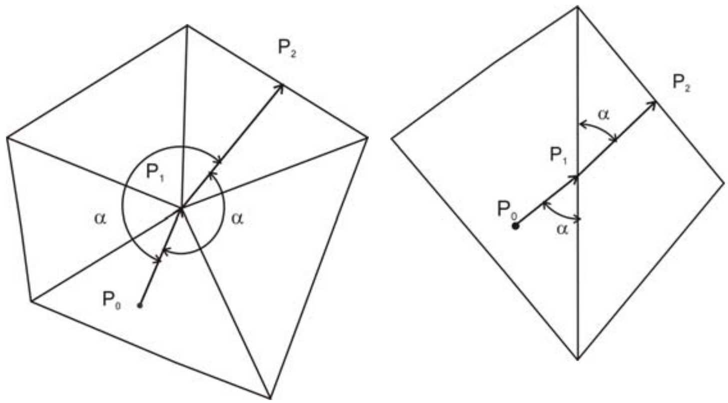


Figure 2. Discrete geodesic computation

2.3 Algorithm for the shortest path

Ing. Zábranský adjusted the above-mentioned algorithm for the shortest path problem in his thesis in June 2005 [8]. He defined the shortest path problem as the boundary-value problem: Given two points s and t, find the discrete geodesic λ which satisfies:

$\lambda(0) = s$
$\lambda(\text{length}(\lambda_{st})) = t$
$\lambda'(0) = \mathbf{v}$
$\text{length}(\lambda_{st}) = \min$

The problem consists in how to find the initially direction for path to pass through the finaly point. The algorithm of Mr. Zábranský chooses the initial direction randomly and runs iteratively. After c. 500 iterations, it chooses the curve that approximates best the shortest path between *s* and *t*. The relative error is about 0.01 after 200 iterations.

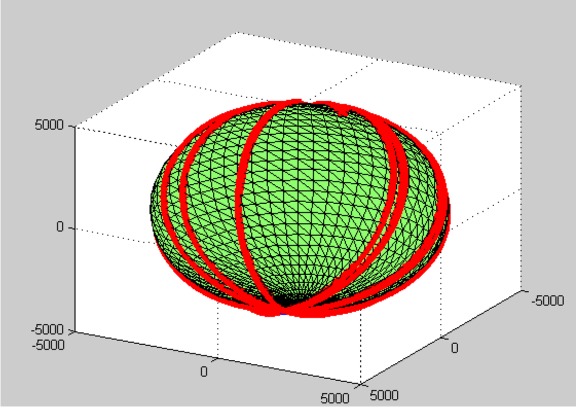


Figure 3. Demonstration of Zábranský's algorithm for 15 iterations.

3 My proposal for acceleration of the algorithm

The algorithm of Mr. Zábranský does not work very effectively. I suggested and implemented a modified algorithm that works much more accurately and faster. In 3.1 the modified algorithm is introduced and in 3.2. the results are presented.

3.1 The modified algorithm

In contrast to Mr. Zábranský I choosed the initial vector v as the difference of the points t and s: $v = t - s$. In the next iterations the vector is changed by a small angle to the both sides from v. The shortest path is found sooner and is more accurately than in the algorithm of Mr. Zábranský.

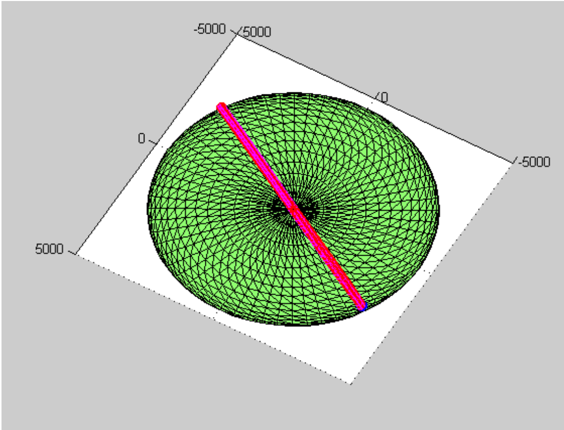


Figure 4a. Demonstration of the modified algorithm for 10 iterations.

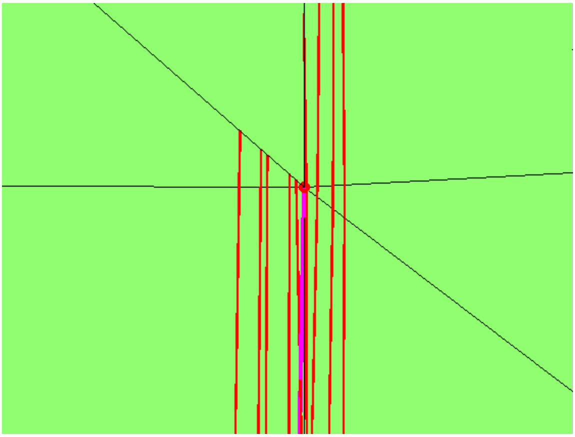
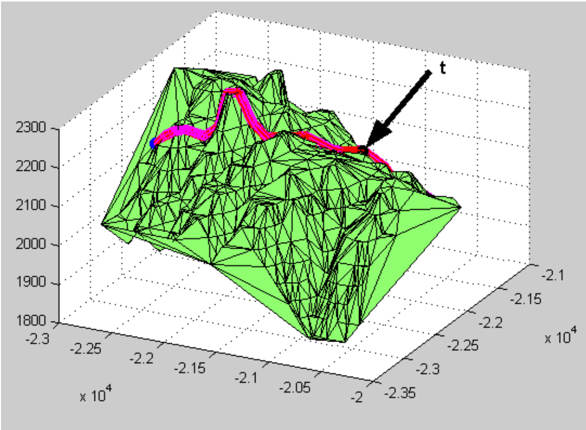


Figure 4b. Demonstration of the modified algorithm for 10 iterations - zoom

3.2 Results

I tested the algorithms on two surfaces: a sphere (3480 triangles) and a non-convex model of terrain (1225 triangles). The algorithm works precisely over the sphere. The demonstration for 10 iterations is displayed at the figure 4. The relative error is less than 0.0001 after 50 iterations.



.

Figure 5a. Demonstration of the modified algorithm on a non-convex surface. The algorithms works precisely.

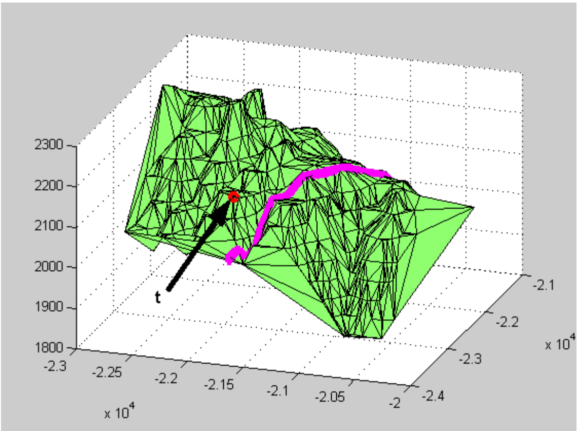


Figure 5b. Demonstration of the modified algorithm on a non-convex surface. Right: The relative error is 0.14

The algorithm does not work too accurately on the non-convex model in some cases. There is an example where the shortest path was found precisely (figure 5a) and an example where the found shortest path distinguishes too much from the exactly shortest path (figure 5b).

4 Conclusion and future work

The time-complexity of algorithm is *O(kn),* where *k* is the number of chosen directions and *n* the number of triangles. In comparison with the other algorithms this algorithm is simple and fast. The main disadvantage of this algorithm is that the path does not pass through the second point *t* exactly.

In the future work I want to improve the algorithm for non-convex surfaces. By interconnecting this algorithm with the algorithm of unfolding I hope to eliminate the main disadvantage: the shortest path should pass the second point exactly.

References

[1] J. Chen, Y. Han: Shortest paths on a polyhedron. *Int. J. Computat. Geom. Appl*. 6, 1996, s.127-144.

[2] S. Kapoor: Efficient computation of geodesic shortest paths. *Proceedings of 31st ACM Symposium on Theory of Computing. ACM*, New York, 1999, s. 770-779.

[3] Kumar, Ravi et al.: Geodesic curve computations on surfaces. *Computer Aided Geometric Design*. May 2003, 20, 2, s. 119-133.

[4] D. Martínek, L. Velho, P. C. Carvalho: Geodesic paths on triangular meshes. *Proceedings of the XVII Brazilian Symposium on Computer Graphics and Image Processing IEEE*, 2004.

[5] J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou: The discrete geodesic problem. *SIAM J. Comput.* 16, 1987, s. 647-668.

[6] A. Porazilová: *Metody dekompozice geometrických objektů, Diplomová práce.* Západočeská univerzita, Plzeň, 2004.

[7] M. Sharir, A. Schorr: On shortest paths in polyhedral spaces. *SIAM J. Comput.* 15, 1986, s. 193-215.

[8] J. Zábranský: *Triangulace povrchů a úlohy na nich, Diplomová práce.* Západočeská univerzita, Plzeň, 2004.