Computing Efficiently Shortest Paths for Degenerate Metrics

Marc Espie
LIENS, ENS, 45 rue d'Ulm, 75230 Paris, France
Currently on leave at Stanford, Robotics Laboratory, CA 94305 USA

Abstract
This paper classifies some metrics according to their length-related properties, expanding this theoretical basis into a practical and efficient algorithm for computing L_1-shortest paths in general plane regions.

Many problems have "shortest path" as a key word, but all of them do not involve the same notions. This paper deals with a degenerate case. For a given norm on a vector space, there usually exists exactly one shortest path between any two points, but for some quirky norms—the L_1-norm being a good example—this is not the case. Indeed, for almost any two points, the number of shortest paths is infinite in that case. It is suitable to call such norms "degenerate". A close study of them provides us with a satisfying geometric characterization of the induced shortest paths. The key notion is the generalization of convexity to arbitrary metrics, due to Menger.

For computational purposes, the existence of a wide class of possible shortest paths can be very useful, because an algorithm can choose the easiest to compute. We further examine the two-dimensional case for the L_1-norm. The resulting algorithm is reasonably simple and has an optimal complexity in the worst case: given an arrangement of k regions described by n monotone curves, the algorithm first performs a preprocessing in $O(n \log n)$ time, using $O(n)$ space, then computes a shortest path between any two points in time $O(k \log k + n)$.

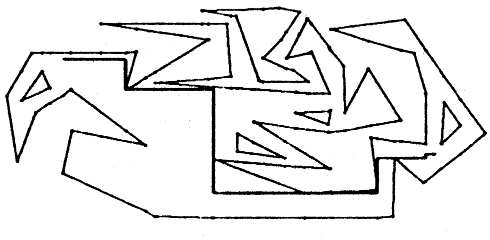

Fig. 1. A sample region, and a rectilinear shortest path.

1–Shortest paths and degenerate metrics

1–1 The notion of a shortest path

Definition: *Length, Shortest paths*

Consider a metric space (E, d) together with a path $f: [a, b] \to E$. A polygonal line Λ underlying f is defined as a sequence of points $\Lambda = (\lambda_0, \ldots, \lambda_n)$ such that there exists $a \leqslant x_0 \leqslant x_1 \leqslant \cdots \leqslant x_n \leqslant b$ verifying $\lambda_i = f(x_i)$. We denote by $\lfloor f \rfloor$ the set of such lines.

The length of Λ is the positive number

$$l(\Lambda) = \sum_{i=1}^{n} d(\lambda_{i-1}, \lambda_i)$$

and **the length of** f itself is the lowest upper bound of the lengths of all polygonal lines underlying it. That is,

$$L(f) = \sup_{\Lambda \in [f]} l(\Lambda).$$

A shortest path between two points a and b is obviously a path whose length is minimal amidst all the paths joining a and b.

This definition is suited for our purposes, because it involve no explicit condition of differentiability. Although in the Euclidean case curves of finite length turn out to be differentiable almost everywhere, no such property is true in the general case. It is actually easy to construct curves of finite L_1-length that are nowhere differentiable.

1–2 Metric Convexity

In order to build shortest paths, a discrete notion of convexity, due to Menger, is very useful. Consider a triplet of points a, b, c in a metric space. The point c is said to sit between a and b if $d(A,C) + d(C,B) = d(A,B)$ and furthermore c is distinct from a and from b.

A subset P of E is called metrically convex if : for every couple of points a, b in P, there exists a point c in P sitting between a and b.

MENGER'S THEOREM

Let P be a metrically convex, complete subset of a metric space E. For any two points a, b of P, there exists a shortest path f between a and b remaining within P. Furthermore, f is a metric segment, that is: f is isometric to a segment of $\mathbf{R}$.

Conversely, metric segments are metrically convex.

Let (E, ν) be a normed metric space and consider the unit ball of ν, which is a closed, convex, symmetric set. As for every convex set, its points are either extremal or non extremal points. Furthermore, since this ball is symmetric, extremal points come in pairs, corresponding to directions of the vector space. For instance, the plane L_1-norm admits two extremal directions: the horizontal and vertical directions, and shortest paths for that norm are essentially curves monotone for these directions; the plane Euclidean norm admits every direction as an extremal direction and shortest paths are straight lines, i.e., curves monotone with respect to every direction.

In the n-dimensional case though, the notion of extremal point is not fine enough to characterize a given norm. The following generalization is suitable though.

Definition: *Extremal Set*

Let S be a subset of a convex set C. This subset is **an extremal set** if: for every point a of S, a can not be obtained as a linear combination of any two points of $C \setminus S$.

Any convex set has got a nice decomposition in extremal sets. For the unit ball of a norm, this decomposition is closely related to metrically convex sets, hence to shortest paths.

THEOREM : CELL DECOMPOSITION OF CONVEX SETS

Let C be a convex set in a vector space of finite dimension n. There exists one and only one cell decomposition $\{D_i\}$ of C verifying the following properties.

- Each D_i is convex.
- Each D_i is an extremal set.
- Each point of C is an interior point of exactly one D_i.

For a given set of extremal directions D_i and a point a, the associated extremal cone is the pencil of straight lines going through a, of directions belonging to D_i.

THEOREM : CHARACTERIZATION OF CONVEX SETS

Let (E, n) be a finite dimensional vector space, of cellular decomposition D_i. Let P be a closed subset of E. Then P is convex if and only if every extremal cone associated to any point of P and any extremal set D_i cuts P through a connected component.

So topologically, shortest paths are defined by the cellular decomposition of the norm. Furthermore, almost every convex has a trivial decomposition: all its boundary points are extremal, so the generic case reduces to the Euclidean case. For degenerate norms though, the situation is much more intricate. Fortunately, we can still rely upon the simple notion of metrically convex sets to build algorithms.

2–The algorithm

2–1 First principles

The use of metric convexity in an algorithm relies on the following scheme: cut the starting region in convex subregions, perform a combinatorial exploration of the resulting graph, and use geometric properties to find a path in that graph which corresponds to an actual shortest path, then build this path. The key points are the following:

- Locality: two points in the same convex region can be linked by a shortest path remaining within that region. We don't have to consider other regions to compute elementary paths.
- Combinatorial simplicity: there exists a shortest path between any two points which does not cross the same convex region twice. So we just have to consider simple paths in the graph, which allows for interesting simplifications.

This is a very general scheme. In that paper, we focus on the two dimensional case of the L_1-metric. A pleasant surprise awaits us: since we didn't take any differentiability assumption, the class of regions that we can process is very general. Instead of polygonal regions, it is perfectly natural here to consider a region bounded by piecewise monotone curves. We denote by n the number of monotone pieces, and by k the number of holes of the region. The first step of the algorithm is very simple.

Proposition:

Using a sweep-line algorithm, it is possible to cut the region in $O(n)$ convex subsets, using $O(n \log n)$ time and $O(n)$ space.

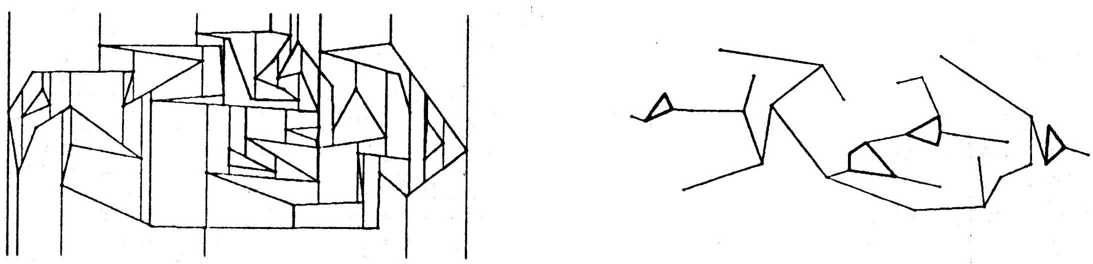

Fig. 2. Decomposition in convex sub-regions

Figures 1 and 2 show a polygonal region and a sample decomposition. Instead of a graph of regions, it seems more practical to consider a "dual" graph, where each region is represented by an edge of the graph, so that vertices of that graph correspond to the vertical lines cutting the region. The principle of combinatorial simplicity applies to that graph as well.

2–2 Analysis of the graph

The size of the graph of regions is $O(n)$, which is big for practical applications. A more significant parameter for combinatorial properties would be the number of holes k. Since we are only interested in simple paths, we can trim the graph down to this size.

- Discard simply-connected components. Two points of the graph belong to the same simply-connected component if there is exactly one simple path joining them. This is an equivalence relation and it is actually well-known: two adjoining vertices belong to the same simply-connected component if the edge between them is an isthmus.

- Eliminate chains. A chain is a path whose internal vertices have degree exactly two. Obviously, any simple path cannot turn back in the middle of a chain.
- Take cut points into account. It turns out that eliminating simply-connected components can introduce vertices of arbitrarily high degree in the graph. Fortunately, such vertices are also cut-points. A close study of that property enables us to consider only subgraphs whose vertices have bounded degree for our search.

Proposition:

The graph of regions is reducible to a simpler graph of size $O(k)$ in time $O(n)$. Every simple path of this graph corresponds to exactly one simple path of the original graph, and the correspondence can be computed in $O(n)$ time.

This simpler graph has not a bounded degree, but for two given points, the search of a simple path can be narrowed to a subgraph of bounded degree at no additional cost.

Figure 2 shows a connected component of the graph of regions. Thick edges are the edges remaining in the quotient graph.

2–3 Geometric paths

In the L_1 metric, a special trick allows for a very efficient computation of shortest paths.

Convex regions are bounded by vertical segments and to obtain a shortest path, it is enough to obtain elementary shortest paths between these segments. That can be done in $O(s)$ time, where s is the size of that region. Now, gluing together such paths is very simple: adding the vertical segment joining their tips does the trick.

The final step of the algorithm is a variant of Dijkstra's classical algorithm: each edge of the final graph corresponds to a chain of convex regions, with an associated shortest path length and path tips. In contrast with the classical Dijkstra scheme, where vertices don't hold any information, we have to glue paths together at each vertex—or to trudge through simply connected regions at cut-points. Taking these modifications into account, the final search for a combinatorial path is an $O(k \log k)$ step. Finally, the actual computation of the path takes $O(n)$ time.

2–4 Optimality and implementation concerns

There exist simply connected regions with shortest paths which can't be described with less than n monotone curves, so the query-time is optimal. Using Chazelle's results on polygonal triangulation, the theoretical time of the preprocessing can be reduced to $O(n)$, though the author expresses some doubts as to the practicality of the process...

In the general case, the graph of regions can be any planar graph of degree 3 with any positive weight on the edges, so any improvement depends heavily upon a better algorithm for finding combinatorial shortest paths in such graphs.

From a pratical standpoint, the implementation of the algorithm is a bit tedious, but straightforward. It exhibits a very good numerical stability, the only tricky part being the sweep-line—which has been thoroughly studied in the literature.

Bibliography

H. Federer, Geometric Measure Theory, *Springer-Verlag, 1970*

B. Chazelle, Polygon Triangulation, *Proceedings of FOCCS 1990*

H. Busemann, Spaces with distinguished Geodesics, *M. Dekker, New York, 1987*

Preparata and Shamos, Computational Geometry, *Springer-Verlag,1985*

L. Blumenthal and K. Menger, Studies in Geometry, *W.H. Freeman and Cie, 1970*

P.J. De Rezende, D.T. Lee, Y.F Wu, Rectilinear Shortest Paths in the Presence of Rectangular Barriers, *Journal of discrete and Computational Geometry, Vol 4-1, 1989*