Fréchet Distance Between Simple Polygons

Kevin Buchin*† Maike Buchin*† Carola Wenk‡

Abstract

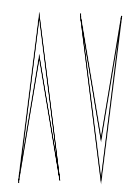
We show that for computing the Fréchet distance between simple (planar) polygons it suffices to choose an arbitrary triangulation of the one polygon and to map the triangulated polygon to the other polygon such that diagonals of the triangulation are mapped to shortest paths in the other polygon. Using this we give a polynomial time algorithm for computing the Fréchet distance between simple polygons. In particular, if one polygon is convex the Fréchet distance of the polygons equals the Fréchet distance of their boundary curves.

The Fréchet distance $\delta_{\mathrm{F}}(f,g)$ between two surfaces $f,g:[0,1]^2 \to \mathbb{R}^d$ is defined as

$$\inf_{\sigma:[0,1]^2\to[0,1]^2} \max_{t\in[0,1]} d(f(t), g(\sigma(t))) \ ,$$

where σ ranges over all orientation-preserving homeomorphisms and $d(.,.)$ is the Euclidean distance. The Fréchet distance between two open or closed polygonal curves of complexity m and n can be computed using a dynamic programming approach in $O(mn \log mn)$ or $O(mn \log^2 mn)$ time, respectively, see [AG95]. Computing the Fréchet distance between polygonal surfaces is NP-hard [God98] and semi-computable [AB05].

In this extended abstract we consider the case that f and g are two simple planar polygons in $\mathbb{R}^2$ For these instead of considering homeomorphisms between the parameter spaces we can instead consider homeomorphisms directly between the polygons because they are homeomorphic. The first question that comes to mind is: Does the Fréchet distance of polygons equal the Fréchet distance of their boundary curves? The figure gives two example polygons for which, when placed on top of each other, the Fréchet distance between the boundary curves is small (roughly the width of the polygons) however the Fréchet distance between the polygons is large (half the height of the polygons).



In the following we sketch how the Fréchet distance between two simple planar polygons can be computed in polynomial time. Details will be given in the full version of this extended abstract.

Lemma 1 (Shortest Paths Lemma)**.** *Given two simple polygons P, Q, a triangulation T of P and a homeomorphism $\sigma : P \to Q$. Then there is a map $\sigma' : P \to Q$ which fulfills*

1. *σ' is a limit of homeomorphisms $P \to Q$*
2. *σ' maps diagonals of T to shortest paths Q and is piecewise linear inside triangles (without introducing interior vertices)*
3. *σ' is "at least as good as σ", i.e., $max_{t\in P} d(t,\sigma'(t)) \le max_{t\in P} d(t,\sigma(t))$*

Proof sketch. The proof is based on incrementally simplifying $\sigma(e)$ for any diagonal $e \in T$. Note that since σ' is piecewise linear, $max_{t\in P} d(t,\sigma'(t)) = max_{t\in E} d(t,\sigma'(t))$, where $t \in E$ denotes all points lying on edges in the edge set E of T (which includes both the boundary and the diagonals of T). □

We immediately get the following corollaries:

Corollary 2. *The Fréchet distance between simple polygons equals $\inf_{\sigma'} \max_{t\in E} d(t,\sigma'(t))$ where E is the edge set of an arbitrary ¡triangulation of P and σ' ranges over all homeomorphisms from the boundary of P to the boundary of Q which are extended to T by mapping the diagonals of T to the shortest paths between the boundary vertices.*

*Freie Universität Berlin, Institute of Computer Science, Takustr. 9, 14195 Berlin, Germany
†This research was supported by the Deutsche Forschungsgemeinschaft within the European graduate program 'Combinatorics, Geometry, and Computation' (No. GRK 588/2).
‡University of Texas at San Antonio, Dept. of Computer Science, 6900 N. Loop 1604 West, San Antonio, TX 78249-0667, USA

Corollary 3. *The Fréchet distance between two simple polygons, of which one polygon is convex, equals the Fréchet distance between their boundary curves.*

Proof. We triangulate the possibly non-convex polygon and map these diagonals to shortest paths in the convex polygon. But the shortest paths in the convex polygon are also diagonals and the distance between diagonals equals the maximal distance of its endpoints. Therefore it suffices to compute the distance between the boundary curves. □

Theorem 4. *Given $\epsilon > 0$ and two simple polygons P, Q with m and n vertices, respectively, we can decide whether $\delta_F(P, Q) < \epsilon$ in time $O(n^3 m^4)$.*

Proof. We sketch an algorithm for the decision problem:

Algorithm DecideFréchet(P,Q)

Input: Simple polygons P, Q, $\epsilon > 0$ **Output:** Is $\delta_F(P, Q) < \epsilon$?

1) Compute a triangulation T of P.
2) Compute the DFSD with its reachability structure of the boundary curves of P, Q.
3) Compute all hour glasses between boundary segments in Q.
4) For all single free space diagrams: Search for a feasible path σ that satisfies $\delta_{\mathrm{F}}(\text{diagonal}, \text{shortest path}) < \epsilon$ for all positionings of diagonals by σ.

A triangulation of P can be computed in $O(n)$ time. Our algorithm makes use of the algorithm of [AG95] for computing the Fréchet distance between closed curves. The algorithm of [AG95] computes a *double free space diagram (DFSD)*, which is the free space diagram of f concatenated f with g. Then it computes a *reachability structure* which divides the boundaries of the DFSD into $O(mn)$ intervals which encode combinatorially different reachability paths. In 2) we compute the DFSD and the reachability structure in $O(mn \log mn)$ time.

An hour glass is a geometrical structure introduced by Guibas and Hershberger [GH87] which encodes all shortest paths between two diagonals in a polygon. In step 3), all hour glasses between boundary segments of Q are constructed in $O(m \log m)$ time.

A *single* free space diagram is a regular free space diagram of f and g with the starting point of f positioned at one of the $O(mn)$ interval boundaries (which were computed in step 2)). In step 4), for each single free space diagram we exploit the nested structure of the diagonals of T in P by using a dynamic programming algorithm to compute all placements of diagonals, for each diagonal using the reachability structure and the hour glasses in order to check whether $\delta_{\mathrm{F}}(\text{diagonal}, \text{shortest path}) < \epsilon$. This can be done in $O(m^3 n^2)$ time per single free space diagram, hence $O(m^4 n^3)$ in total. □

To go from deciding to computing we do the same as for curves in [AG95], i.e., compute a set of critical values and perform parametric search on them, which adds another log-factor to the runtime. As critical values we use the ones for the boundary curves plus all Fréchet distances between diagonals in P and hour glasses in Q.

References

[AB05] Helmut Alt and Maike Buchin. Semi-computability of the Fréchet distance between surfaces. In *Proc. 21st European Workshop on Computational Geometry*, 2005.

[AG95] Helmut Alt and Michael Godau. Computing the Fréchet distance between two polygonal curves. *Internat. J. Comput. Geom. Appl.*, 5:75–91, 1995.

[GH87] L. J. Guibas and J. Hershberger. Optimal shortest path queries in a simple polygon. In *SCG '87: Proceedings of the third annual symposium on Computational geometry*, pages 50–63, New York, NY, USA, 1987. ACM Press.

[God98] Michael Godau. *On the complexity of measuring the similarity between geometric objects in higher dimensions.* PhD thesis, Freie Universität Berlin, Germany, 1998.