

Computing the Fréchet distance with shortcuts is NP-hard

Citation for published version (APA):

Buchin, M., Driemel, A., & Speckmann, B. (2013). Computing the Fréchet distance with shortcuts is NP-hard. 43-46. Abstract from 29th European Workshop on Computational Geometry (EuroCG 2013), Braunschweig, Germany.

Document status and date:

Published: 01/01/2013

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Computing the Fréchet distance with shortcuts is NP-hard

Maike Buchin*

Anne Driemel†

Bettina Speckmann*

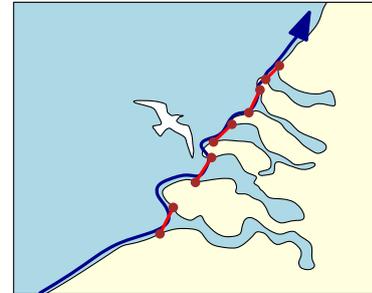
Abstract

We study the *shortcut Fréchet distance*, a natural variant of the Fréchet distance that allows us to take shortcuts from and to any point along one of the curves. We show that, surprisingly, the problem of computing the shortcut Fréchet distance exactly is NP-hard. Furthermore, we give a 3-approximation algorithm for the decision version of the problem.

1 Introduction

Measuring the similarity of two curves is an important problem which occurs in many applications. A popular distance measure, that takes into account the continuity of the curves, is the Fréchet distance. Imagine walking forwards along the two curves simultaneously. At any point in time, the two positions have to stay within distance ε . The minimal ε for which such a traversal is possible is the Fréchet distance. In general, the Fréchet distance can be computed by the algorithm of Alt and Godau [1] in $O(n^2 \log n)$ time. Despite its versatility, the Fréchet distance has one serious drawback: it is a bottleneck distance. Hence it is quite sensitive to outliers, which are frequent in real world data sets. To remedy this Driemel and Har-Peled [3] introduced a variant of the Fréchet distance, namely the *shortcut Fréchet distance*, that allows shortcuts from and to any point along one of the curves. The shortcut Fréchet distance is then defined as the minimal Fréchet distance over all possible such shortcut curves.

The shortcut Fréchet distance automatically cuts across outliers and allows us to ignore data specific “detours” in one of the curves. Hence it produces significantly more meaningful results when dealing with real world data than the classic Fréchet distance. Consider the following example. Birds are known to use coastlines for navigation, e.g., the Atlantic flyway for migration. However, when the coastline takes a “detour”, like a harbor or the mouth of a river, the



bird ignores this detour, and instead follows a shortcut across. See the example of a seagull in the figure, navigating along the coastline of Zeeland while taking shortcuts between the islands. Using the shortcut Fréchet distance, we can detect if the trajectory of the bird is similar to the coastline. The shortcut Fréchet distance can be interpreted as a partial distance measure. Note that a different notion of a partial Fréchet distance was developed by Buchin et al. [2].

Definitions. A *curve* T is a continuous mapping from $[0, 1]$ to \mathbb{R}^2 , where $T(t)$ denotes the point on the curve parameterized by $t \in [0, 1]$. Given two curves T and B in \mathbb{R}^2 , the *Fréchet distance* between them is

$$d_{\mathcal{F}}(T, B) = \min_{f: [0,1] \rightarrow [0,1]} \max_{\alpha \in [0,1]} \|T(f(\alpha)) - B(\alpha)\|,$$

where f is an orientation-preserving reparameterization of T . We call the line segment between two arbitrary points $B(y)$ and $B(y')$ on B a *shortcut* on B . Replacing a number of subcurves of B by the shortcuts connecting their endpoints results in a shortcut curve of B . Thus, a *shortcut curve* is an order-preserving concatenation of non-overlapping subcurves of B that has straight line segments connecting the endpoints of the subcurves. Our input are two polygonal curves: the *target curve* T and the *base curve* B . The *shortcut Fréchet distance* $d_{\mathcal{S}}(T, B)$ is now defined as the minimal Fréchet distance between the target curve T and any shortcut curve of the base curve B .

Results. In this paper we study the complexity of computing the shortcut Fréchet distance. Driemel and Har-Peled [3] described approximation algorithms for the shortcut Fréchet distance in the restricted case where shortcuts have to start and end at input vertices. Specifically, they gave a $(3 + \varepsilon)$ -approximation algorithm for the vertex-restricted shortcut Fréchet distance between c -packed polygonal curves that runs

*Department of Mathematics and Computer Science, TU Eindhoven, the Netherlands, m.e.buchin@tue.nl and speckman@win.tue.nl. Supported by the Netherlands Organisation for Scientific Research (NWO) under project no. 612.001.106. and 639.022.707.

†Department of Information and Computing Sciences Utrecht University, the Netherlands, a.driemel@uu.nl. Supported by the Netherlands Organisation for Scientific Research (NWO) under project no. 612.065.823.

in $O(c^2 n \log^3 n)$ time for two c -packed polygonal curves of complexity n . Firstly, we outline how to combine the algorithmic layout of Driemel and Har-Peled [3] with a line stabbing algorithm of Guibas et al. [4] to obtain a 3-approximation algorithm for the decision version of the general shortcut Fréchet distance which runs in $O(n^3 \log n)$ time. This result is described in the full version of this paper.

Secondly, we show that, surprisingly, in the general case, where shortcuts can be taken at any point along a curve, the problem of computing the shortcut Fréchet distance exactly is NP-hard. An important observation is that the reachable free space of the matchings may fragment into an exponential number of components. We use this fact in our reduction together with a mechanism that controls the sequence of free space components that may be visited. In this abstract we describe the construction, we prove the correctness in the full version.

2 NP-Hardness

Assume an instance of SUBSET-SUM is given to us as n positive integers $S = \{s_1, s_2, \dots, s_n\}$ and a positive integer σ . Recall that the problem is to decide whether there exists an index set I , such that $\sum_{i \in I} s_i = \sigma$. We now describe a construction of a target curve T and base curve B such that there exists a shortcut curve of the base curve that is within Fréchet distance 1 to the target curve if and only if there exists a subset of S of which the total sum is σ .

Basic Idea. The base curve has several horizontal edges within distance 1 of the target curve. A feasible shortcut curve has to visit a well-defined subsequence of these edges. Any possible visiting sequence will encode a different subset of S . Let I be an index set which defines such a subset $S' \subseteq S$, we call $s_i = \sum_{1 \leq j \leq i, j \in I} s_j$ the i th *partial sum* of S' . The incremental partial sums encoded by a shortcut curve are encoded on certain edges of the base curve by the particular point where the shortcut visits the edge. We can restrict the solutions to the Fréchet problem to have this specific form by using what we call projection centers. These are certain points on the target curve, which have to be visited by any curve that is within Fréchet distance 1 to the target curve. Intuitively a shortcut of a feasible shortcut curve has to start and end at particular edges of the base curve and intersect a projection center in between. Hence we can think of the shortcut as a projection. The construction is such that a shortcut that enters a gadget will have two edges of the base curve available as possible destinations. The corresponding projections will then cascade through the projection centers of the gadget and are bundled again on the last edge of the gadget, where they have a certain distance to each other, which encodes one of the input values.

General layout and notation. We denote with H_0, H_1, H_{-1} and H_α the horizontal lines at 0, 1, -1 and α . All relevant vertices of the construction lie on these lines (see Figure 1) and hence it is usually sufficient to specify their x -coordinates. We slightly abuse notation by denoting the x -coordinate of a point and the point itself with the same variable, albeit using a different font.

The target curve consists entirely of edges that lie on the x -axis. In Figure 1, the curve drawn below the x -axis illustrates the topology of the target curve. Clearly all feasible shortcut curves have to lie within the hippodrome of radius 1 around the target curve. The i th gadget defines a subcurve T_i . The vertices of T_i , except for the initialization and the terminal gadget, are defined by the parameters $c_j^{(i)}$ for $1 \leq j \leq 4$. These are $(c_1^{(i)} + 1, 0), (c_1^{(i)} - 1, 0), (c_2^{(i)} + 1, 0), (c_2^{(i)} - 1, 0), (c_3^{(i)} + 1, 0), (c_3^{(i)} - 1, 0), (c_4^{(i)} + 1, 0), (c_4^{(i)} - 1, 0)$ in this order. Thus, the edges of the target curve are generally running in positive x -direction, except for some edges of length two, which are centered at the points $(c_j^{(i)}, 0)$. We call these points *projection centers*. The construction of the base curve is such that any feasible shortcut curve has to go through the projection centers. In particular, this is enforced by the fact that we place all edges of the base curve at distance at least 2 away from the projection centers.

The base curve has relevant edges $e_j^{(i)}$, for $1 \leq j \leq 7$ and $0 \leq i \leq n$, where j defines the order along the base curve. These edges lie on the horizontal lines H_1, H_{-1} and H_α at 1, -1 and $\alpha \in (0, 1)$. We call these edges *docking edges*, since they are the edges visited by the feasible shortcut curves. The docking edges run in negative x -direction. The remaining edges of the base curve are outside the hippodrome, except for *connector edges*, which vertically connect to docking edges on H_α and run in positive y -direction.

Global variables. The construction uses four global variables $\alpha \in (0, 1), \beta > 0$, and $\delta > 0$. The parameter α is besides 1 and -1 the y -coordinate of the horizontal lines that support the docking edges. The parameter β controls the minimal horizontal distance between docking edges that lie in between two consecutive zones. The function of the parameter δ is two-fold. Firstly, it is the minimum difference of two partial sums. This can be ensured by scaling the instance by δ , such that $s_i/\delta \geq 1$ for $1 \leq i \leq n$ and $\sigma/\delta \geq 1$. Secondly, we choose δ sufficiently large to ensure that a feasible shortcut curve cannot visit any edges other than docking edges and only in the prescribed visiting order.

Encoding of a solution. A shortcut curve B_\diamond of the base curve encodes a subset $S' \subseteq S$ as follows: The value s_i is included in S' if and only if B_\diamond visits $e_1^{(i)}$. Any feasible shortcut curve B_\diamond also encodes an

approximation of its incremental partial sums in the distance between the point where B_\diamond visits the edge $e_7^{(i)}$ to the endpoint of this edge $a_7^{(i)}$. By choosing the global parameter δ carefully, we can ensure that two distinct partial sums have a minimum difference that exceeds the approximation error. By construction of the terminal gadget any feasible shortcut curve has to visit $e_7^{(n)}$ at a point that is in distance $\sigma + \delta$ to the point $a_7^{(i)}$. This implies that only shortcut curves that encode a subset that sums to σ can be feasible.

Construction of the gadgets. We now describe the part of the construction of the gadgets that is specific to the instance of the problem. That is, we give exact choices of the coordinates of the two curves. The construction is incremental. Given the endpoints of edge $e_7^{(i-1)}$, as defined by the $(i-1)$ th gadget and the value s_i , we describe how to construct the subcurves of the intermediate gadget G_i . We describe the initialization and the terminal gadget afterwards. Since all relevant vertices of the base and target curve lie on horizontal lines as indicated in Figure 1, we need to choose only their x -coordinates. The construction goes through several rounds of fixing the position of the next projection center and then projecting an endpoint $a_j^{(i)}$ of one edge to obtain the endpoint $a_{j+1}^{(i)}$, $a_{j+2}^{(i)}$, or $a_{j+3}^{(i)}$ of another edge. The endpoint $b_j^{(i)}$ is projected in the same way. Thus, we obtain the first point of one edge by projecting the last point of another and the other way around.

The detailed construction is described in the full version of the paper. Here, we only describe how to pick the first and the last projection center. From G_{i-1} , we are given the values of $a_7^{(i-1)}$ and $b_7^{(i-1)}$. Let $\ell = b_7^{(i-1)} - a_7^{(i-1)}$ and let $h_i = b_7^{(i-1)} + \beta + \ell$. We choose c_1^i as the x -coordinate where the line through $(h_i, -\alpha)$ and $(b_7^{(i-1)}, -1)$ passes through H_0 . We obtain $a_j^{(i)}$ and $b_j^{(i)}$ for $1 \leq j \leq 6$ from the subsequent projections through the constructed projection centers as shown in the figure. Now, let $p_i = a_6^{(i)} - s_i$. We choose c_4^i as the x -coordinate where the line through $(p_i, 1)$ and $a_5^{(i)}$ passes through H_0 . And finally we project the points $a_5^{(i)}$, $b_5^{(i)}$, $a_6^{(i)}$ and $b_6^{(i)}$ through the last projection center c_4^i onto H_{-1} . We then choose $a_7^{(i)}$ as the minimum of the obtained x -coordinates and $b_7^{(i)}$ as the maximum of the obtained x -coordinates.

In this manner we obtain the docking edges $e_j^{(i)}$ for $1 \leq j \leq 7$. We connect $e_7^{(i-1)}$ to $e_1^{(i)}$ using edges that lie outside the hippodrome. Similarly we connect the remaining edges in the order of j using vertical connector edges for the edges lying on H_α and otherwise edges that lie outside the hippodrome.

Initialization. We place the first vertex of the target curve at $(a_0^{(0)}, 0) = (0, 0)$ and the first vertex of the

base curve at $(a_0^{(0)}, 1) = (0, 1)$. The base curve then continues to the left on H_1 while the target curve continues to the right on H_0 . G_0 has one projection center $(c_1^{(0)}, 0)$, we define it by $c_1^{(0)} = \delta + 2$. Then we define $e_7^{(0)}$ such that $a_0^{(0)}$ projects onto the center of this edge and such that the projection is in distance δ to both endpoints. That is, we define $a_7^{(0)} = c_1^{(0)} + 2$ and $b_7^{(0)} = c_1^{(0)} + 2\delta + 2$. Now, the next gadget G_1 can be constructed as described above.

Terminal gadget. We choose the very last projection center by setting $c_1^{(n+1)} = b_7^{(n)} + 2$. Let $p_\sigma = (a_7^{(n)} + \delta + \sigma)$ and project the point $(p_\sigma, -1)$ through this projection center onto H_1 to obtain a point $(a_\sigma, 1)$. We finish the construction by letting both the target curve and the base curve end on a vertical line at a_σ . The target curve ends on H_0 approaching from the left, while the base curve ends on H_1 approaching from the right.

Proof Idea. Consider the following construction of a shortcut curve that encodes a given subset $S' \subseteq S$. We start in $B(0)$, and subsequently project through all projection centers. In the intermediate gadget for s_i , we visit $e_1^{(i)}$ if $s_i \in S'$, otherwise we visit $e_2^{(i)}$. Finally, we choose $B(1)$ as the last vertex of our shortcut curve. We claim that this curve is feasible if and only if S' is a solution. This can be proven by a repeated application of the intercept theorem. Note that this curve visits any edge of the base curve in at most one point. Clearly not all feasible shortcut curves have this property. However, they have to be approximately monotone by the construction of the target curve. This helps us to bound the error in the encoding of the partial sums.

Acknowledgements. We thank Maarten Löffler for insightful discussions on the topic of this paper.

References

- [1] H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. *Int. J. of Comp. Geometry & Applications*, 5:75–91, 1995.
- [2] K. Buchin, M. Buchin, and Y. Wang. Exact algorithm for partial curve matching via the Fréchet distance. In *Proc. 20th ACM-SIAM Symp. on Discrete Algorithms*, pages 645–654, 2009.
- [3] A. Driemel and S. Har-Peled. Jaywalking your dog – computing the Fréchet distance with shortcuts. In *Proc. 23rd ACM-SIAM Symp. on Discrete Algorithms*, pages 318–337, 2011.
- [4] L. J. Guibas, J. Hershberger, J. S. B. Mitchell, and J. Snoeyink. Approximating polygons and subdivisions with minimum link paths. In *Proc. 2nd Int. Symp. on Algorithms*, pages 151–162, 1991.

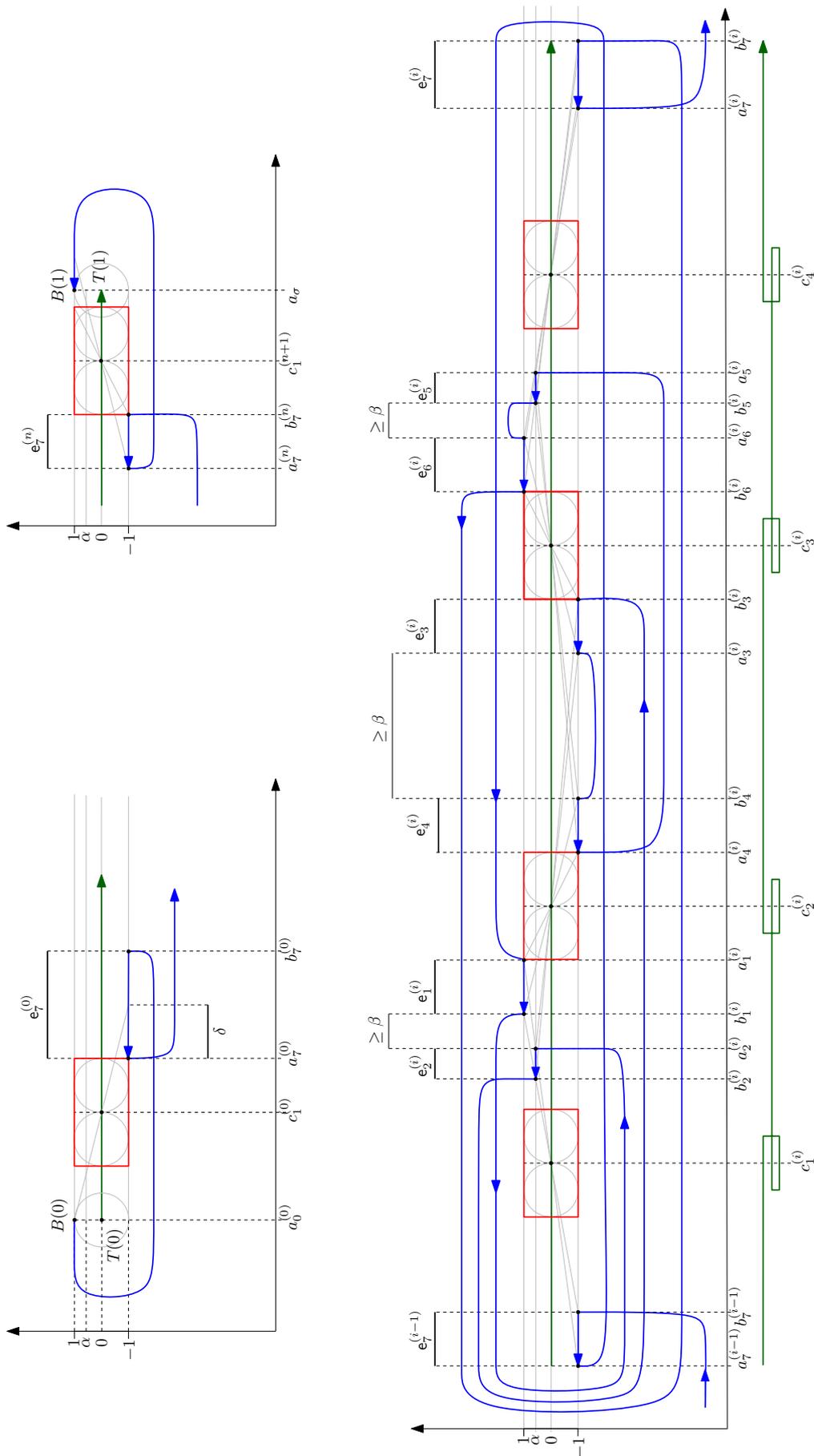


Figure 1: Initialization gadget, Terminal gadget and intermediate gadgets G_i with global parameters α and β . The target curve is shown in green, the base curve in blue. For the sake of presentation the lengths of the docking edges have been assumed smaller.