Length-Preserving Matching Between Closed Curves

Erfan Hosseini Sereshgi ✉
Department of Computer Science, Tulane University, United States

Maarten Löffler ✉
Department of Information and Computing Sciences, Utrecht University, The Netherlands

Frank Staals ✉
Department of Information and Computing Sciences, Utrecht University, The Netherlands

Carola Wenk ✉
Department of Computer Science, Tulane University, United States

1 Introduction

Considering length when comparing geometric shapes is of crucial importance in many fields such as object recognition, pattern recognition, image stitching, biomedical imaging, robotics and automation, 3D modeling and reconstruction and geographic information systems (GIS).

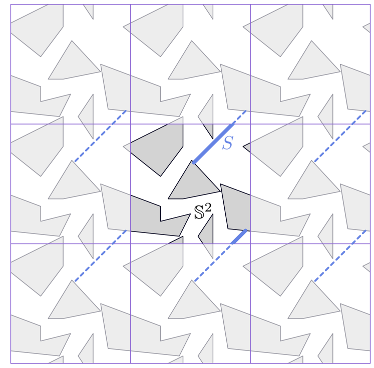
At the same time, length has proven to be a notoriously unstable quantity in many applications, as it can depend on the resolution of the representation of a shape [8]. Combined with the fact that in digital datasets shapes are often stored as point sequences, most existing distance measures for shapes are either explicitly discrete in nature (such as, e.g., the Hausdorff distance [4]), or if they can handle truly continuous data (such as the Fréchet distance [6]), they explicitly *disregard* the length of features. While dozens of distance measures have been introduced over the years, we are only aware of a single attempt to explicitly consider length: the *length-sensitive Fréchet similarity* by Buchin et al. [5]. However, it also enforces restrictive order-preserving constraints which might not be necessary in some comparisons. In this work, we propose a continuous, one-to-one matching approach that preserves length but allows for partial order flexibility.

Problem Definition: Let $P, Q : \mathbb{S}^1 \to \mathbb{R}^2$ be two closed polygonal curves of equal length (assumed wlog to be one). We wish to find a partition of P and Q into k subcurves $P_1, .., P_k$ and $Q_1, .., Q_k$, respectively, so that matched subcurves have the same length, and we can simultaneously traverse them at unit speed while staying within distance δ. We model this as a problem in the free space diagram: Let $\delta > 0$ be a given matching threshold. We consider the subdivision of the two dimensional parameter space $\mathbb{S}^2$ into free and forbidden space. A point $(\alpha, \beta) \in \mathbb{S}^2$ is in the free space iff $\|P(\alpha) - Q(\beta)\|_1 \leq \delta$. We wish to find an ordered set $S_1, .., S_k$ in $\mathbb{S}^2$ of k line segments such that:

1. The horizontal projections of the segments form intervals $X_1, .., X_k$ with disjoint interiors that together cover all of P;
2. The vertical projections of the segments form intervals $Y_1, .., Y_k$ with disjoint interiors, that together covers all of Q; and
3. Every segment has slope 1 or -1.

The problem is thus to find a 1-covering solution for P and Q, meaning our objective is to find a set of k segments (with slopes 1 and -1) that maximizes the length ℓ of the covered intervals and checks whether this length is one. In particular, this abstract focuses on matching closed curves, demonstrating a possible NP-hardness proof for the general case of k, with a polynomial-time solution for $k = 2$ under L_1.

From Torus to the Plane. With closed curves as inputs, the resulting free space diagram effectively wraps around a torus. To illustrate this structure, consider $\mathcal{D} \subseteq \mathbb{S}^2$ be the planar unfolding of nine copies of $\mathbb{S}^2$ in a 3×3 grid. Any segment s in a solution Ψ is represented by a set of line segments (dashed) in $\mathcal{D}$.



▶ **Observation 1.** *Every segment S in the free space of $\mathbb{S}^2$ is represented by at least one line segment in the free space of $\mathcal{D}$.*

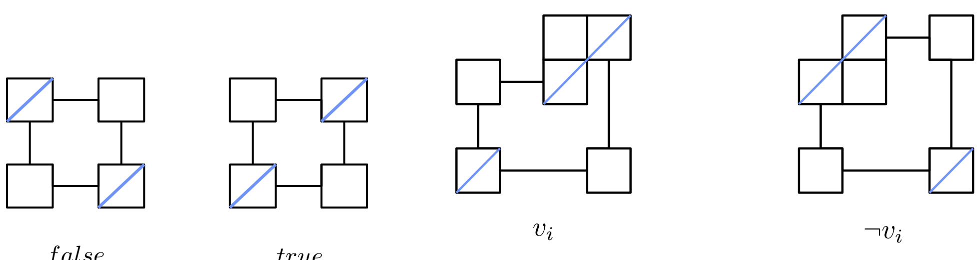
2 NP-Hardness for General k

▶ **Theorem 2.** *The problem of finding if there exist k slope-one and slope-negative-one segments in the free-space such that their vertical and horizontal projections fully cover both axes without any overlaps is NP-hard even if the segments have a fixed length.*

Proof Sketch. We prove the NP-hardness of the non-constant k-segment free-space covering problem by reducing it from Max-2SAT [7]. Max-2SAT involves finding an assignment to a CNF formula (with clauses of up to two literals) that maximizes satisfied clauses.

Given a CNF formula with m variables and t clauses, we map it to a free-space diagram with $4m + 4t$ candidate slope-one segments, represented as square boxes. As shown in Figure 1, each variable is modeled as a cycle of four squares, where half of the boxes are filled based on the truth value assigned to the variable. We add two squares to the cycle for each appearance of a variable. For each clause, two squares from different cycles share a corner, with additional squares added to ensure full axis coverage without overlap.

The construction ensures that satisfying s clauses in the Max-2SAT problem corresponds to having $k \geq 2(m + t) - s$ segments in the free-space diagram. Conversely, covering the free-space with k segments guarantees $s \geq 2(m + t) - k$ satisfied clauses in the original Max-2SAT formula. The reduction is completed in polynomial time. ◀



(a) $True$ and $false$ states of a variable.

(b) Two satisfied single-variable clauses. On the left, v_i is *true* while on the right v_i is *false*.

Figure 1 Some examples of the blocks used in the structure.

Theorem 2 offers key insights into the computational complexity of computing an optimal length-preserving matching between two closed curves for general k. While it doesn't prove NP-hardness, it strongly suggests it, as similar problems have shown this to be indicative [2,3]. However, constructing curves that match a given free-space diagram remains a difficult problem, only recently starting to be understood [1].

3 Covering The Free-space With $k = 2$ Segments

For each vertex v of the forbidden space in $\mathcal{D}$, consider the maximal segment M_v of slope 1 through v. These segments induce a trapezoidal decomposition Λ^1 of $\mathcal{D}$ (See Figure 2). Λ^{-1} can be constructed similarly by drawing slope -1 segments.

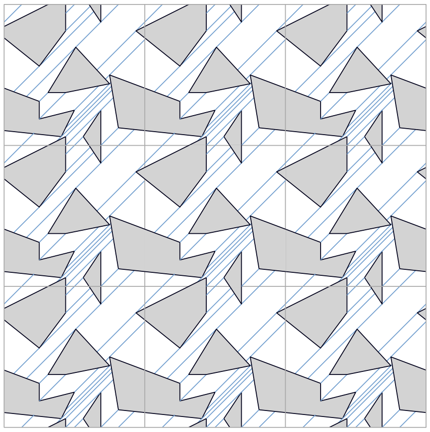


Figure 2 A maximal length slope 1 segment corresponds to maximal segment M_v of a vertex v in the trapezoidal decomposition Λ^1.

▶ **Lemma 3.** *The decompositions Λ^1 and Λ^{-1} can be constructed in $O(n \log n)$ time where n is the complexity of the free space.*

Proof Sketch. Simple sweepline algorithm: We can compute the decomposition for Λ^1 by employing a slope 1 sweep line originating from the top left corner of $\mathcal{D}$ and terminating at the bottom right corner. The free space vertices serve as event points, sorted in the direction of the sweep line. As the sweep line progresses, we maintain a balanced binary tree of free space edges intersected by the sweep line, enabling the efficient computation of the closest free-space edge to the event point. This portion of the sweep line forms the edge between two adjacent trapezoids.

The algorithm consists of two primary operations: After sorting $O(n)$ event points which requires $O(n \log n)$ time. The sweepline starts at the first event point. For each event point, we perform a closest-edge query, which takes $O(\log n)$ using the binary tree. Hence, the total runtime complexity is $O(n \log n)$. Such approach can also be applied for Λ^{-1}. ◀

We now discuss a solution for $k = 2$ segments. First, we observe that if both segments have the same slope, they belong to the same trapezoid and they are in fact aligned into a single segment, and we reduce to the case $k = 1$ (This case is not complex and can be solved in $O(n \log n)$). Therefore, we now assume that our solution $\Psi = \{B, R\}$ consists of two segments, with $B = \overline{b_1 b_2}$ having slope 1 and $R = \overline{r_1 r_2}$ having slope -1. As shown in Figure 3, if there are two segments R, B that *cover* $\mathbb{S}^2$ then there are a total of five unique cases:

(a) R goes through an obstacle vertex v and r_2 lies on an obstacle edge, and b_1 lies on an obstacle edge.
(b) R goes through an obstacle vertex v, and both r_1 and r_2 lie on obstacle edges.
(c) R goes through an obstacle vertex v, r_2 lies on an obstacle edge, and b_1 lies on an obstacle vertex v'.

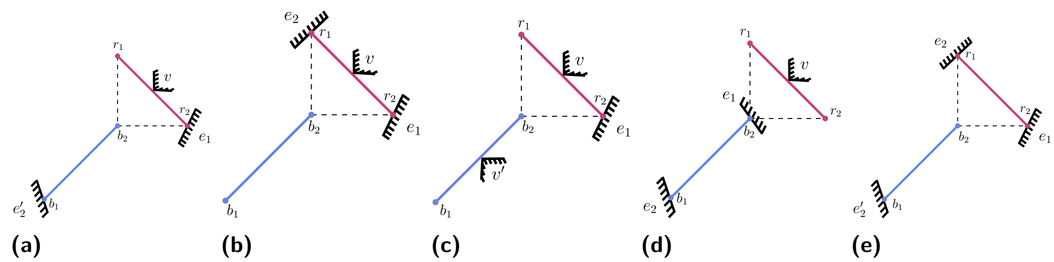


Figure 3 Cases for $k = 2$. B and R are shown in blue and red, respectively. Obstacle vertices and edges are in black.

(d) R goes through an obstacle vertex v, and both b_1 and b_2 lie on obstacle edges.
(e) r_1 and r_2 lie on obstacle edges, and b_1 lies on an obstacle edge.

Due to symmetry, the roles of R and B are interchangeable in these cases.

▶ **Theorem 4.** *We can solve the $k = 2$ case in $O(n^2)$ time.*

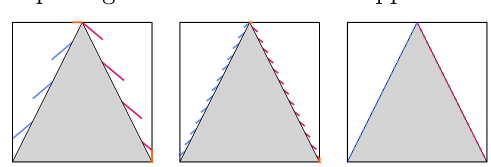
Proof Sketch. Based on Lemma 3, constructing the decomposition Λ^1 and Λ^{-1} takes $O(n \log n)$ time. We evaluate each case by selecting a pair of trapezoids (one from each decomposition) and testing in constant time whether they can form a valid solution, with the red segment nestled in one trapezoid and the blue segment in the other. Having $O(n)$ trapezoids in each decomposition, results in the total runtime of $O(n^2)$. ◀

It is worth noting that the algorithm described above can be optimized for the first three cases. By bounding the red (or symmetrically, blue) segment with a vertex and an edge, the corresponding blue (or red) trapezoid is fixed, eliminating the need to check all possible trapezoids.

4 Conclusion and Discussion

We introduced length-preserving matching and presented algorithms to compute it for $k = 1$ and $k = 2$, while demonstrating NP-hardness for the general case. In addition to potential runtime improvements for specific $k = 2$ cases, more research is required to explore the problem for larger values of k and if the current general algorithm for $k = 2$ can be effectively scaled up for this purpose.

A promising avenue for future research involves exploring the scenario where k approaches infinity. As illustrated in the figure, increasing k and thereby decreasing the size of each segment leads to greater coverage, even enabling the matching of entire curves as $k \to \infty$ which is not possible with a fixed number for k. While several



properties suggest the potential for a polynomial algorithm in this case, further investigation is needed, and our current definition requires modification to accommodate this setup. Furthermore, since the length of each segment approaches zero, it adds further ambiguity to the concept, necessitating a more nuanced understanding. Finally, an alternative research direction would be to explore a similar measure that averages the distances of all matched parts, rather than focusing solely on the maximum distance.

References

1 Hugo A. Akitaya, Maike Buchin, Majid Mirzanezhad, Leonie Ryvkin, and Carola Wenk. Realizability of Free Spaces of Curves. In Satoru Iwata and Naonori Kakimura, editors, *34th International Symposium on Algorithms and Computation (ISAAC 2023)*, volume 283 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 3:1–3:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: `https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ISAAC.2023.3`, `doi:10.4230/LIPIcs.ISAAC.2023.3`.

2 Hugo Alves Akitaya, Maike Buchin, Leonie Ryvkin, and Jérôme Urhausen. The k-Fréchet Distance: How to Walk Your Dog While Teleporting. In Pinyan Lu and Guochuan Zhang, editors, *30th International Symposium on Algorithms and Computation (ISAAC 2019)*, volume 149 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 50:1–50:15, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: `https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ISAAC.2019.50`, `doi:10.4230/LIPIcs.ISAAC.2019.50`.

3 Boris Aronov, Anne Driemel, Marc van Kreveld, Maarten Löffler, and Frank Staals. Segmentation of trajectories for non-monotone criteria. In *Proc. 24th Symposium on Discrete Algorithms*, pages 1897–1911, 2013. URL: `http://knowledgecenter.siam.org/0236-000092/1`.

4 T. Birsan and D. Tiba. One hundred years since the introduction of the set distance by Dimitrie Pompeiu. In F. Ceragioli, A. Dontchev, H. Futura, K. Marti, and L. Pandolfi, editors, *System Modeling and Optimization*, pages 35–39, Boston, MA, 2006. Springer US.

5 Kevin Buchin, Brittany Terese Fasy, Erfan Hosseini Sereshgi, and Carola Wenk. On length-sensitive Fréchet similarity. In *Algorithms and Data Structures: 18th International Symposium, WADS 2023, Montreal, QC, Canada, July 31 – August 2, 2023, Proceedings*, page 208–231, Berlin, Heidelberg, 2023. Springer-Verlag. `doi:10.1007/978-3-031-38906-1_15`.

6 M. Maurice Fréchet. Sur quelques points du calcul fonctionnel. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 22(1):1–72, 1906.

7 M.R. Garey, D.S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976. URL: `https://www.sciencedirect.com/science/article/pii/0304397576900591`, `doi:https://doi.org/10.1016/0304-3975(76)90059-1`.

8 Benoit Mandelbrot. How long is the coast of Britain? Statistical self-similarity and fractional dimension. *Science*, 156(3775):636–638, 1967. `doi:10.1126/science.156.3775.636`.