Minimum Backward Fréchet Distance

Amin Gheibi[*] [†]
School of Computer Science, Carleton University, Ottawa ON, Canada
agheibi@scs.carleton.ca

Anil Maheshwari[†]
School of Computer Science, Carleton University, Ottawa ON, Canada
anil@scs.carleton.ca

Jörg-Rüdiger Sack[*] [†]
School of Computer Science, Carleton University, Ottawa ON, Canada
sack@scs.carleton.ca

Christian Scheffer[‡]
Department of Computer Science, Westfälische Wilhelms-Universität Münster, Germany
Christian.Scheffer@uni-muenster.de

ABSTRACT

We propose a new measure to capture similarity between polygonal curves, called the *minimum backward Fréchet distance*. It is a natural optimization on the weak Fréchet distance, a variant of the well-known Fréchet distance. More specifically, for a given threshold ε, we are searching for a pair of walks for two entities on the two input curves, T_1 and T_2, such that the union of the portions of backward movements is minimized and the distance between the two entities, at any time during the walk, is less than or equal to ε. Our algorithm detects if no such pair of walks exists. This natural optimization problem appears in many applications in Geographical Information Systems, mobile networks and robotics. We provide an exact algorithm with time complexity of $\mathcal{O}(n^2 \log n)$ and space complexity of $\mathcal{O}(n^2)$, where n is the maximum number of segments in the input polygonal curves.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.3.5 [**Computational Geometry and Object Modeling**]: Geometric Algorithms; I.5.3 [**Clustering**]: Similarity Measures

General Terms

Algorithms

Keywords

Fréchet Distance, Polygonal Curves Similarity, Optimization

1. INTRODUCTION

Measuring the similarity between two polygonal curves is among the fundamental problems of computational geometry. It poses challenges and is of high interest both from a practical and theoretical point of view. In the context of Geographical Information Systems (GIS) the similarity of movement patterns, modeled by polygonal curves, has a variety of applications. These include animal behaviour, human movement, traffic management, sports scene analysis, and movement in abstract spaces [9, 10, 11]. In [13, p. 352] it is stated that for curve matching, a global approach (e.g. via the Fréchet distance) achieves a better accuracy than a local approach (e.g., via the Hausdorff distance or the distance proposed in [8]). The reason is, a global matching takes continuous and global parametrization of the considered curves into account. For this reason, the Fréchet distance is a widely used and established tool to measure and formalize the similarity between polygonal curves [10, 11]. Its global character makes it a more "suitable" [2;, 13, p. 859] measure for continuous curves.

In the well-known dog-leash metaphor, the Fréchet distance can be described informally as follows: suppose a man walks his dog, while both have to move on their own curves from the starting point to the ending point on their respective curves. The Fréchet distance is the minimum needed length for the dog's leash, if the person and the dog are not allowed to move backwards on the pair of curves. An often used variant of the standard Fréchet distance is the weak Fréchet distance (or non-monotone Fréchet distance) [1]. In this version, the objective is to minimize the required leash length while backward movements of arbitrary length are allowed at each point during the walk. In GIS, the weak Fréchet distance has e.g., been used in the context of map matching [2, 13].

In robotics, the weak Fréchet distance is related to a measure

*This research is partially supported by High Performance Computing Virtual Laboratory and SUN Microsystems of Canada.
†This research is supported by Natural Sciences and Engineering Research Council of Canada.
‡This research was supported in part by DAAD grant 57052155.

SIGSPATIAL'14, November 04 - 07 2014, Dallas/Fort Worth, TX, USA

http://dx.doi.org/10.1145/2666310.2666418

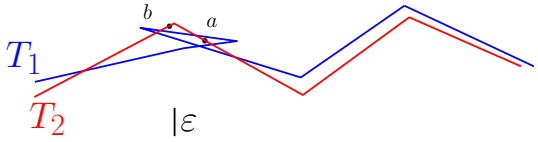

Figure 1: Moving backward from a to b allows to walk on T_1 and T_2 and keeping the distance between moving objects less than ε during the walk.

known as ring-width, e.g., studied in [1, 12]. In the ring-width problem, the input is a closed polygon, P, and two half-lines, h_1 and h_2. The starting point of h_1 and h_2 is on the boundary of P and they do not intersect each other or P at any other point. The objective is to find the minimum width ring such that it is possible to move P through the ring, starting from h_1 and ending at h_2. This problem was solved for the first time by Goodman et al. [12] and an alternative easier solution was provided by Alt and Godau [1], based on the weak Fréchet distance.

It is known that the standard Fréchet distance is sensitive to outliers. To address this, new measures together with their algorithmic solutions have been introduced that are less sensitive to outliers and obtain a better curve-matching (see e.g., [3, 5]). For two input polygonal curves, Buchin et al. [3] studied maximizing the length of matched subcurves which are in ε Fréchet distance of each other. They solved the problem exactly when the Fréchet distance is calculated under the L_1 or L_∞ norms. For the standard L_2 norm, Carufel et al. [5] studied how to minimize the length of subcurves of two polygonal curves that need to be removed to achieve a given Fréchet distance. Recently, Buchin et al. [4] studied a variant of the problem that shortcuts replace the removed subcurves. These shortcuts are considered for matching during the computation of Fréchet distance. They showed that this problem is NP-hard.

Brakatsoulas et al. [2] pointed out that the choice between the weak Fréchet distance and the standard Fréchet distance depends on the application. Figure 1 illustrates an example of two polygonal curves that appear similar and, with a small amount of backward movement, the curves are indeed in weak ε Fréchet distance (note that their standard Fréchet distance is large and thus they would not be considered to be similar under that measure). While in the standard Fréchet distance no backtracking is allowed, in the weak Fréchet distance, the moving entities are allowed to backtrack on the curves. A natural question arises: what is the minimum length required for such backward movements to achieve a particular Fréchet distance? This optimization problem is solved here. Note that a simultaneous minimization of the length of backward movements and the Fréchet distance, are conflicting objectives. Increasing (resp., decreasing) the maximum leash length decreases (resp., increases) the minimum required length of all backward movement portions. Hence, we search for an optimal solution, i.e., walks that minimizes the backward movements, with a fixed input upper bound on the leash length.

In the language of the dog-leash metaphor, the considered variant of the Fréchet distance of this paper is motivated as follows: imagine that an upper bound for the allowed leash length is given as an input parameter. This could be a maximum allowed distance between the two moving entities so that they can quickly reunite, in case of an emergency. Now, we are searching for walks on the given curves, such that the entire distances traveled are minimized. If the Fréchet distance between the input curves is not larger than the input threshold, then the optimal solution is clearly equal to the sum of the lengths of the both input curves (since no backward movement is necessary). In this situation, a corresponding optimal pair of walks can be computed via the approach from [1], i.e., a pair of walks, which realizes the Fréchet distance. If this is not the case, an optimal solution, or more precisely any valid solution, forces the man and/or his dog to move backwards, such that the sum of the traveled distances is larger than the lengths of the two input curves. Our objective is to minimize the length of the portions that are traveled backwards, while guaranteeing that the leash length stays below the input threshold.

The structure of the paper is as follows. In Section 2, we discuss the preliminaries and define the problem formally. Then, in Section 3, we explain a polynomial time algorithm to solve the problem exactly. Afterward, in Section 4, we improve the time and space complexity of the algorithm to $\mathcal{O}(n^2 \log n)$ and $\mathcal{O}(n^2)$ respectively. At the end, we conclude the paper and pose some open questions.

2. PROBLEM DEFINITION

In this section, we first state some preliminary concepts. Then, we will define the minimum backward Fréchet distance problem formally. A geometric path in $\mathbb{R}^2$ is a sequence of points in the Euclidean space, $\mathbb{R}^2$. A discrete geometric path, or a polygonal curve, is a geometric path, sampled by a finite sequence of points (i.e., vertices), which are connected by line segments (i.e., edges) in order. Let $T_1 : [0, n] \to \mathbb{R}^2$ and $T_2 : [0, m] \to \mathbb{R}^2$ be two polygonal curves of complexity (number of segments) n and m respectively. W.l.o.g., assume that $m \leq n$. A parameterization of $[0, n]$ is a continuous function $f : [0, 1] \to [0, n]$, where $f(0) = 0$ and $f(1) = n$ hold ($[0, 1]$ is a time interval). If f is non-decreasing, then the parameterization is monotone. The weak Fréchet distance, $\delta_w(T_1, T_2)$, is defined as Formula 1, where $d(.,.)$ is the Euclidean distance and f and g are two parameterization of $[0, n]$ and $[0, m]$ respectively, which are not necessarily monotone. However, for the standard Fréchet distance, the two parameterization must be monotone.

$$\delta_w(T_1, T_2) = \inf_{f,g} \max_{t\in[0,1]} d(T_1(f(t)), T_2(g(t))) \tag{1}$$

For a parameterization f, let $\mathcal{B}_f \subseteq [0, 1]$ be the closure of the set of times in which $f(t)$ is decreasing (i.e., the movement is backward). For a pair of parameterizations we define its quality by Formula 2, where $\|.\|$ is the Euclidean length.

$$Q_{f,g}(T_1, T_2) := \int_{t\in\mathcal{B}_f} ||T_1(f(t))||dt + \int_{t\in\mathcal{B}_g} ||T_2(g(t))||dt \tag{2}$$

We formally define the minimum backward Fréchet distance as follows. For a pair of polygonal curves, T_1 and T_2, and a

given leash length, ε, we are looking for a pair of optimal parameterization, (f, g), as defined in Formula 3. We consider only matchings that induce matched pairs of points within distance less than or equal ε.

$$Q^{\varepsilon}(T_1, T_2) = \inf_{f,g} Q_{f,g}(T_1, T_2) \tag{3}$$

The basic structure to decide whether the Fréchet distance between two polygonal curves is upper bounded by a given ε, is the free-space diagram [1]. For two polygonal curves, T_1 with n vertices and T_2 with m vertices, and two corresponding parameterizations, f and g, the free-space is defined formally by Formula 4.

$$F_\varepsilon = \{(t_1, t_2) \in [0,1] \times [0,1] \mid d(T_1(f(t_1)), T_2(g(t_2))) \leq \varepsilon\} \tag{4}$$

The free-space diagram is the rectangle $[0,1] \times [0,1]$, partitioned into n columns and m rows. It consists of nm parameter cells $C^{i,j}$, for $i = 1, ..., n$ and $j = 1, ..., m$, whose interiors do not intersect with each other. The cell $C^{i,j}$ represents the multiplication of two subranges of $[0,1]$ that are mapped to the edge between vertices $T_1(i-1)$ and $T_1(i)$ and the edge between vertices $T_2(j-1)$ and $T_2(j)$. For each parameter cell $C^{i,j}$, there exists an ellipse such that the intersection of the area bounded by this ellipse with $C^{i,j}$ is equal to the free-space region of that cell. The union of all cells' free-space builds the free-space (or white-space) of the diagram and is denoted by W. The complement of W is the forbidden-space (or black-space) of the diagram and is denoted by B.

Since we will measure the lengths of the subcurves of T_1 and T_2 directly in the free-space diagram, we stretch and compress the columns and rows of the diagram, such that their widths and heights are equal to the lengths of the corresponding segments. The resulting diagram is called the deformed free-space diagram and is denoted by $\mathcal{F}$. In Figure 2a, two polygonal curves, a leash length, ε, and the corresponding deformed free-space diagram are shown. As the deformed free-space diagram illustrates, to be able to walk on T_1 and T_2 with a leash length less than ε, there must be a backward movement on the polygonal curves. However, the possible walk is not unique. We are looking for a walk that has the minimum backward movement. In this example, the value for the optimal walk is $\delta(t_{11}, t_{12}) + \delta(t_{21}, t_{22})$ where δ is the walking distance between the two points on the corresponding curve.

3. ALGORITHM

In order to solve the minimum backward Fréchet distance problem, we transform it to a shortest path problem on a graph, $\mathcal{G}_v = \langle V, E \rangle$. Consider the forbidden-space, B, as obstacle. We define the visibility inside the free-space, W, with respect to the obstacle. If it is possible to link two points inside W by a line segment such that it does not intersect B, then we say they are visible. The set of vertices, V, of $\mathcal{G}_v$ is the set of all vertices of W (the free-space). The vertices are the intersection points of the ellipses with the boundary of the configuration space cells, in addition to two points, $\mathcal{S}$ and $\mathcal{T}$. The bottom-left corner of the free-space

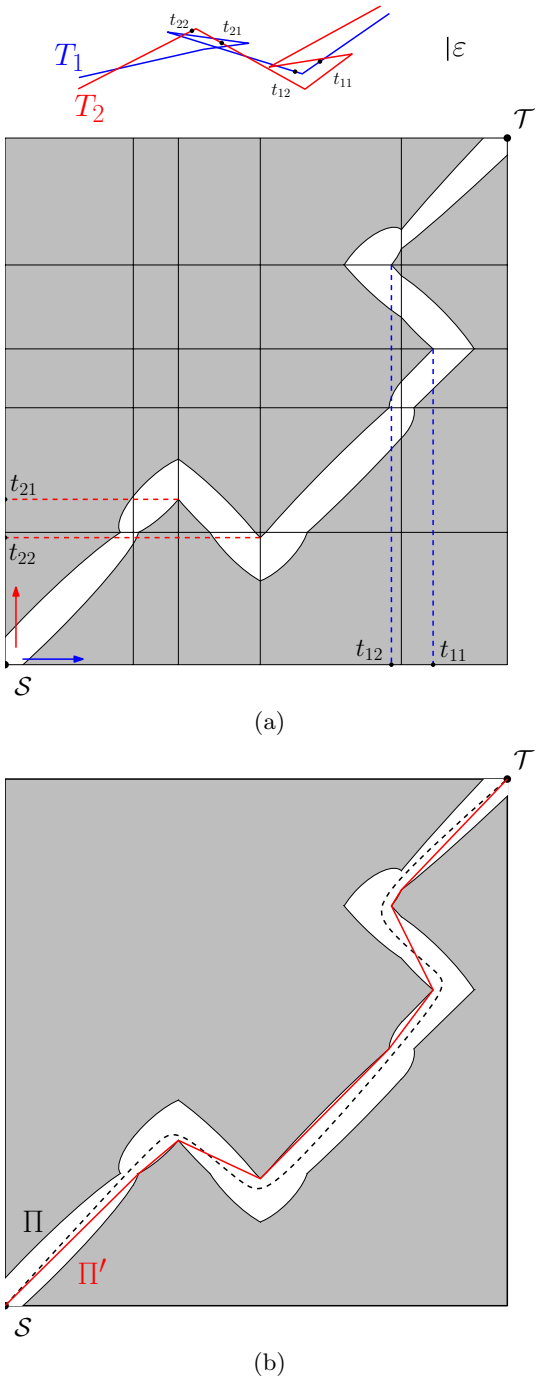

Figure 2: (a) Two polygonal curves, T_1 and T_2, and the leash length, ε, are shown. Also the corresponding deformed free-space diagram is drawn. (b) Two paths in the free-space are drawn: an arbitrary path Π' (black dashed line) and an optimal path $\Pi \subset \mathcal{G}_v$ (red solid line).

diagram $\mathcal{S}$, called the source, corresponds to the starting points of the polygonal curves. Also, the top-right corner of the free-space diagram $\mathcal{T}$, called the target, corresponds to the ending points of the polygonal curves. Each vertex, $v \in V$, has two coordinates, v_x and v_y. Every two visible vertices of W, v_1 and v_2, are linked by two directed edges in E, from v_1 to v_2, $\langle v_1, v_2\rangle$, and vice versa, $\langle v_2, v_1\rangle$. If a directed edge $e = \langle v_1, v_2\rangle \in E$ is xy-increasing (i.e., it is non-decreasing from v_1 to v_2 in both x and y axes), its weight is zero. If e is only x-increasing (resp. y-increasing), its weight is $|v_{1y} - v_{2y}|$ (resp. $|v_{1x} - v_{2x}|$). Otherwise, the weight of e is $|v_{1x} - v_{2x}| + |v_{1y} - v_{2y}|$, which is equal to the L_1-distance between two vertices. If either $\mathcal{S}$ or $\mathcal{T}$ are not in W, then there is no solution for the given leash length. Otherwise, both $\mathcal{S}$ and $\mathcal{T}$ are in $\mathcal{G}_v$ as vertices and we prove that a shortest path from $\mathcal{S}$ to $\mathcal{T}$ gives an optimal walk. Note that in this paper the vertices of the graph are also presented by a point in W. Therefore, a path in the visibility graph could be converted identically to a path in W by connecting its consecutive vertices by a line segment.

OBSERVATION 1. *Let $\Pi : [0,1] \to [0,n] \times [0,m]$ be a path in the free-space diagram, from $\mathcal{S}$ to $\mathcal{T}$. Π is equivalent to a pair of parameterization of the two polygonal curves, $(f : [0,1] \to [0:n], g : [0,1] \to [0:m])$.*

In this paper, a path from $\mathcal{S}$ to $\mathcal{T}$ is denoted by its vertices, $\Pi : \langle \mathcal{S} = p_1, p_2, \ldots, p_k = \mathcal{T}\rangle$. All line segments in Π are directed, $\overrightarrow{p_i p_{i+1}}$, $i = 1, \ldots, k-1$. The length (i.e., cost) of a segment is a function of its direction and is denoted by $|\overrightarrow{p_i p_{i+1}}|$ and defined as follows. If it is xy-increasing, its length is zero. If it is only x-increasing (resp. y-increasing), its length is $|p_{iy} - p_{i+1y}|$ (resp. $|p_{ix} - p_{i+1x}|$). Otherwise, its length is $|p_{ix} - p_{i+1x}| + |p_{iy} - p_{i+1y}|$, which is equal to the L_1-distance between p_i and p_{i+1}. The length of a path, $|\Pi|$, is the sum of the length of its segments. In addition, the notation Π_i is used to denote the sub-path of Π from p_1 to p_i.

In this paper, we use norms in two spaces: (1) the Euclidean space of the input polygonal curves, called the input space, (2) the deformed free-space diagram, called the configuration space. In the input space, we use the Euclidean length of a polygonal curve T and denote it by $||T||$. In configuration space, we use two norms to measure the length of a path Π: the length of a path in the L_1 metric, denoted by $|\Pi|_1$, and the direction-based norm, defined in the last paragraph, denoted by $|\Pi|$.

Lemma 1 is the main lemma in this paper. The corollary of this lemma and Observation 1 is that the corresponding path of an optimal pair of parameterization, called optimal path, is subset of $\mathcal{G}_v$.

LEMMA 1. *For any path $\Pi : \langle \mathcal{S} = p_1, p_2, \ldots, p_{k_1} = \mathcal{T}\rangle$ in W, there is a path $\Pi' : \langle \mathcal{S} = p'_1, p'_2, \ldots, p'_{k_2} = \mathcal{T}\rangle$ in W such that $\Pi' \subset \mathcal{G}_v = \langle V, E\rangle$ and $|\Pi'| \le |\Pi|$.*

PROOF. We describe an algorithm which takes as input the path Π and transform it to a path Π' such that Π' is a path in $\mathcal{G}_v$ between $\mathcal{S}$ and $\mathcal{T}$. The algorithm for constructing Π' is stated in Algorithm 1. At the beginning, Π' contains only $\mathcal{S} = p'_1 = p_1$. Then, Π' is constructed by inserting p'_j, $j = 2$ to k_2, to the tail of Π'.

We prove the correctness of this algorithm by induction. The induction is on i, which is the index of the vertices of Π, $i \in I = \{1, \ldots, k_1\}$.

Algorithm 1 Constructing Π'

Input: The free-space W, A path $\Pi : \langle \mathcal{S} = p_1, p_2, \ldots, p_{k_1} = \mathcal{T}\rangle$
Output: A path $\Pi' : \langle \mathcal{S} = p'_1, p'_2, \ldots, p'_{k_2} = \mathcal{T}\rangle$

```
1: Π' : ⟨S⟩;
2: p'_z = S;
3: for i=2 : k_1 − 1 do
4:     if p'_z p_{i+1} ∈ W then
5:         Continue the loop;
6:     else
7:         Compute the convex chain from p'_z to p_{i+1}, CC_z^{i+1} : ⟨p'_z, q_1, ..., q_c, p_{i+1}⟩;
8:         Insert q_j, j = 1, ..., c, to the tail of Π';
9:         p'_z = q_c;
10: Insert T to tail of Π';
11: return Π';
```

The algorithm maintains the following invariants (before the i-th iteration of the **for** loop in Algorithm 1):

1. $\overrightarrow{p'_z p_i} \in W$ where p'_z is the latest inserted vertex to the tail of Π'.
2. $\Pi'_z \subset \mathcal{G}_v$.
3. $|\Pi'_z| + |\overrightarrow{p'_z p_i}| \le |\Pi_i|$.

We show this by induction on i, the index of the vertices of Π (and the index of the **for** loop in Algorithm 1). The base case of the induction is $i = 2$. For this case, p'_z is equal to $p'_1 = \mathcal{S}$. Obviously, $\overrightarrow{p'_1 p_2} \in W$, $\Pi'_1 \subset \mathcal{G}_v$ and $|\Pi'_1| + |\overrightarrow{p'_1 p_2}| \le |\Pi_2|$ since $\overrightarrow{p'_1 p_2} = \overrightarrow{p_1 p_2}$. The induction hypothesis is that the invariants hold up to the step i (before the i-th iteration of the **for** loop in Algorithm 1). We will prove that they also hold for step $i+1$ (after finishing the i-th iteration of the **for** loop in Algorithm 1).

The main **for** loop in Algorithm 1 has two cases: *a)* $\overrightarrow{p'_z p_{i+1}} \in W$, *b)* $\overrightarrow{p'_z p_{i+1}} \notin W$. In case *a*, p_i is ignored and Π' remains unchanged. In case *b*, first, a convex chain from p'_z to p_{i+1} is constructed. This convex chain is the Euclidean shortest path, directed from p'_z to p_{i+1} (see Figure 3). It is denoted by $CC_z^{i+1} : \langle p'_z, q_1, \ldots, q_c, p_{i+1}\rangle \in W$, where $q_j \in V$, $j = 1, \ldots, c$ and $1 \le c \le n^2$. After computing CC_z^{i+1}, the algorithm inserts q_j, from $j = 1$ to $j = c$, to the tail of Π'. In the remaining, we will prove that for any of the cases the invariants hold.

Each segment in a path could be one of the following types: 1. xy-increasing 2. x-increasing 3. y-increasing 4. neither 1,

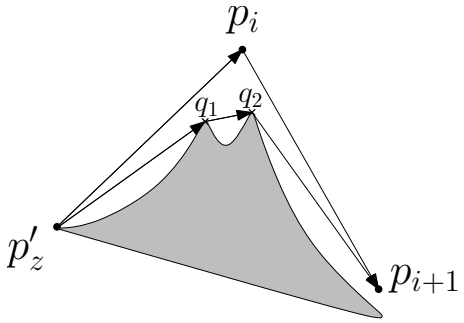

Figure 3: The convex chain from p'_z to p_{i+1} (inside the free-space), $CC_z^{i+1} : \langle p'_z, q_1, q_2, p_{i+1} \rangle$.

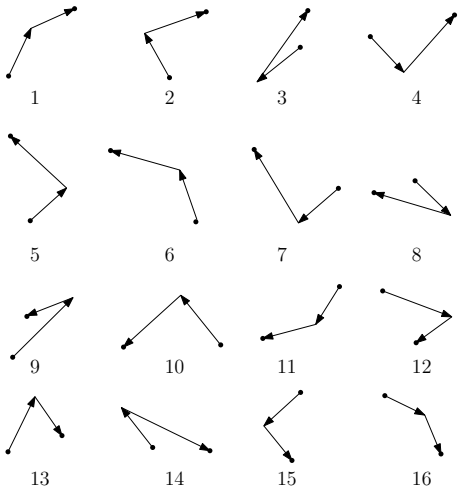

Figure 4: There are 16 cases for the combination of two directed segments.

nor 2, nor 3. Therefore, there are 16 cases for the combination of two segments, $\overrightarrow{p'_z p_i}$ and $\overrightarrow{p_i p_{i+1}}$ (Figure 4). For case a of the **for** loop, when $\overrightarrow{p'_z p_{i+1}} \in W$, invariants 1 and 2 hold, due to the induction hypothesis. To show that the invariant 3 still holds after step i, we go through the 16 cases. For cases 1, 6, 11 and 16, the length of $\overrightarrow{p'_z p_{i+1}}$ is equal to $|\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}|$. By induction hypothesis, we had $|\Pi'_z| + |\overrightarrow{p'_z p_i}| \leq |\Pi_i|$. Now add $|\overrightarrow{p_i p_{i+1}}|$ to both sides of the inequality. Therefore, $|\Pi'_z| + |\overrightarrow{p'_z p_{i+1}}| \leq |\Pi_{i+1}|$. For other 12 cases, $|\overrightarrow{p'_z p_{i+1}}| < |\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}|$. Therefore, $|\Pi'_z| + |\overrightarrow{p'_z p_{i+1}}| < |\Pi'_z| + |\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}| \leq |\Pi_{i+1}|$.

To prove the correctness for case b of the **for** loop, we need some notation. Take every directed segment of a path as a vector from the origin. The angle of a vector, θ, is defined as the angle between that vector (in Cartesian coordinate system) and the positive direction of x-axis. The angles of segments of $CC_z^{i+1} : \langle p'_z, q_1, \ldots, q_c, p_{i+1} \rangle$ are denoted by θ_μ, $\mu = 1, .., c+1$. Based on the convexity of CC_z^{i+1}, the following three properties are easy to prove:

1. CC_z^{i+1} is inside the triangle $p'_z p_i p_{i+1}$.
2. Assume α is the angle of $\overrightarrow{p'_z p_i}$ and β is the angle of $\overrightarrow{p_i p_{i+1}}$. Then, $\alpha \leq \theta_\mu \leq \beta$.
3. The sequence of θ_μ, μ from 1 to $c+1$, is in a sorted order (either increasing or decreasing).

Since CC_z^{i+1} is a convex chain from p'_z to p_{i+1}, invariants 1 and 2 hold. In order to check if the invariant 3 holds for case b, we analogously consider the 16 cases (Figure 4). Here we only show the proofs for two cases of Figure 4 and the proves for the other cases are analogous.

Consider the case when both $\overrightarrow{p'_z p_i}$ and $\overrightarrow{p_i p_{i+1}}$ are only y-increasing (see case 6 in Figure 4). In this case, we have $\frac{\pi}{2} \leq \alpha < \beta < \pi$. Therefore, $\frac{\pi}{2} \leq \theta_\mu < \pi$, $\mu = 1, \ldots, c+1$. Since the sequence of θ_μ is increasing, CC_z^{i+1} is a x-monotone polygonal chain (i.e., any vertical line intersect it in at most one point). Hence, $|CC_z^{i+1}| = |\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}|$. By inductive hypothesis we had $|\Pi'_z| + |\overrightarrow{p'_z p_i}| \leq |\Pi_i|$. Add $|\overrightarrow{p_i p_{i+1}}|$ to both sides of the inequality. We obtain $|\Pi'_z| + |CC_z^{i+1}| \leq |\Pi_i| + |\overrightarrow{p_i p_{i+1}}|$. It follows that $|\Pi'_{z+c}| + |\overrightarrow{q_c p_{i+1}}| \leq |\Pi_{i+1}|$, where q_c is the latest inserted vertex to the tail of Π' and Π'_{z+c} is the sub-path of Π' from index 1 to index $z+c$. The proof for cases 1,11 and 16 is similar.

Consider Case 9, when $\overrightarrow{p'_z p_i}$ is xy-increasing and $\overrightarrow{p_i p_{i+1}}$ decreases in both x and y axes (described in Figure 5). The vertical line that passes through p_{i+1} is denoted by $L_x^\perp$. The horizontal line that passes through p_{i+1} is denoted by $L_y^\perp$. In addition to the three properties mentioned with respect to the convexity of CC_z^{i+1}, the following two properties hold:

4. Any directed segment of CC_z^{i+1} that lies on the left of $L_x^\perp$ is xy-increasing. Therefore, they have a cost zero.
5. Any directed segment of CC_z^{i+1} that lies below $L_y^\perp$ is y-increasing. Therefore, they have a cost zero in the y-dimension.

By Property 4, the cost of CC_z^{i+1} in x-dimension is less than or equal to $|p_{i_x} - p_{i+1_x}|$. Analogously, by Property 5, the cost of CC_z^{i+1} in y-dimension is less than or equal to $|p_{i_y} - p_{i+1_y}|$. In this case, since $\overrightarrow{p'_z p_i}$ is xy-increasing, its cost is zero. Therefore, $|CC_z^{i+1}| \leq |p_{i_x} - p_{i+1_x}| + |p_{i_y} - p_{i+1_y}| = |\overrightarrow{p_i p_{i+1}}| = |\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}|$. By inductive hypothesis, $|\Pi'_z| + |\overrightarrow{p'_z p_i}| \leq |\Pi_i|$. Add $|\overrightarrow{p_i p_{i+1}}|$ to both sides of the inequality. Thus, we obtain $|\Pi'_z| + |CC_z^{i+1}| \leq |\Pi'_z| + |\overrightarrow{p'_z p_i}| + |\overrightarrow{p_i p_{i+1}}| \leq |\Pi_i| + |\overrightarrow{p_i p_{i+1}}|$. It follows that $|\Pi'_{z+c}| + |\overrightarrow{q_c p_{i+1}}| \leq |\Pi_{i+1}|$. Therefore, invariant 3 also holds for this case. The proof for the other remaining cases is similar. □

COROLLARY 1. *There is an optimal path which is a subset of the visibility graph of the free-space diagram, $\mathcal{G}_v$.*

PROOF. Assume Π is an optimal path in the free-space W. If Π is not a subset of $\mathcal{G}_v$, then, by Lemma 1 there is a path, Π' in W such that $\Pi' \subset \mathcal{G}_v$ and $|\Pi'| \leq |\Pi|$. Since Π is an optimal path, $|\Pi'| = |\Pi|$. □

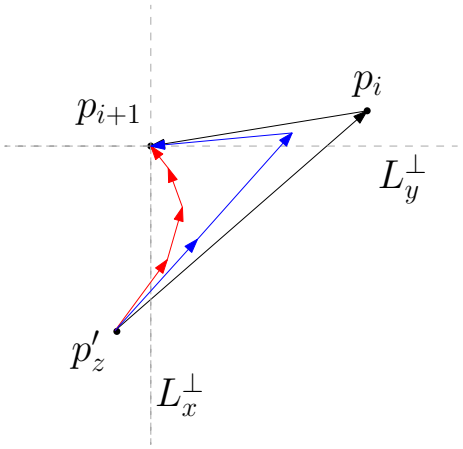

Figure 5: Proof for case 9. If the convex chain (shown by red color) does not go above the horizontal line, $L_y^{\perp}$, which is passing through p_{i+1}, then all of the segments in the chain are y-increasing.

COROLLARY 2. *An optimal pair of parameterization is achievable via a shortest path in the visibility graph, $\mathcal{G}_v$.*

PROOF. Follows directly from Observation 1 and Corollary 1. □

THEOREM 1. *Assume that T_1 and T_2 are two polygonal curves and a leash length, ε, is given. A parameterization of T_1 and T_2 that minimizes the backward movement during the walk can be found in polynomial time and space.*

PROOF. The correctness follows directly from Corollary 2. For the space complexity, observe that the free-space diagram has a complexity of $\mathcal{O}(n^2)$. Therefore, the number of edges of $\mathcal{G}_v$ (and the total space complexity) is upper-bounded by $\mathcal{O}(n^4)$. To find an edge of the visibility graph, a brute force algorithm checks the intersection with all $\mathcal{O}(n^2)$ ellipses. Therefore, the construction of $\mathcal{G}_v$ takes $\mathcal{O}(n^6)$ time. It is possible to find a shortest path in the graph in $\mathcal{O}(|E| + |V| \log |V|) = \mathcal{O}(n^4)$. Therefore, the total time complexity is $\mathcal{O}(n^6)$. Note that if the representing nodes for $\mathcal{S}$ and $\mathcal{T}$ in $\mathcal{G}_v$ are not in a connected component of $\mathcal{G}_v$, then there is no feasible walk with the leash length of at most ε. Therefore, the algorithm halts with the answer of no feasible solution. □

4. IMPROVEMENT

In Corollary 1 of the previous section, we proved that the visibility graph of the free-space, $\mathcal{G}_v$, contains an optimal solution to our problem. As we will show next, it is not necessary to compute the complete visibility graph. In order to find an optimal path, we transform W to a polygonal domain, $\mathcal{D}$. Then, we use the recent algorithm of Chen and Wang [6] to compute an L_1 shortest path in $\mathcal{D}$. More precisely, a polygonal domain, $\mathcal{D}$, with k vertices and h holes is a connected and closed subset of $\mathbb{R}^2$ having h holes whose boundaries consist of $h+1$ simple closed polygonal chains with a total of k vertices.

The process of building $\mathcal{D}$ from the free-space W is as follows. For each cell $C^{i,j}$, find the intersection of the boundary of $C^{i,j}$ with its corresponding ellipse $\phi^{i,j}$. Assume that the intersection points are sorted CCW on the boundary of $\phi^{i,j}$, $\langle v_1, v_2, ..., v_\rho \rangle$. If the part of the boundary of $\phi^{i,j}$ from v_i to v_{i+1}, $i = 1, ..., \rho - 1$, is inside $C^{i,j}$, connect v_i to v_{i+1} by a line segment (Figure 6), otherwise, ignore it. To complete the process of building $\mathcal{D}$ from W, add the line segments of the bounding polygon around the free-space diagram.

A known property of the free-space diagram states that every two adjacent cells in the free-space diagram, $C^{i,j}$ and $C^{i+1,j}$, have the same set of intersection points on the edge shared between them. Therefore, the mentioned process creates conforming polygonal chains. In addition, since $\phi^{i,j}$ is a convex object and the process of building $\mathcal{D}$ from W does not remove any of the vertices of W, the visibility graph of $\mathcal{D}$ is identical to $\mathcal{G}_v$. By Corollary 1, there is an optimal path in $\mathcal{G}_v$. Therefore, there exists an optimal path in $\mathcal{D}$.

LEMMA 2. *Let T_1 and T_2 be two polygonal curves, ε be the leash length, and W be the free-space of the deformed free-space diagram, $\mathcal{F}$. Suppose Π_1 and Π_2 are two paths in W from $\mathcal{S}$ to $\mathcal{T}$. If, in the L_1 norm, $|\Pi_1|_1 \leq |\Pi_2|_1$, then $|\Pi_1| \leq |\Pi_2|$.*

PROOF. Suppose (f_1, g_1) and (f_2, g_2) are the corresponding parameterization pairs of Π_1 and Π_2, respectively. We know that the length of any path in W from $\mathcal{S}$ to $\mathcal{T}$ is at least $||T_1|| + ||T_2||$, where $\|.\|$ is the Euclidean length of a polygonal curve. Therefore, $|\Pi_1|_1 = ||T_1|| + ||T_2|| + 2 * \int_{t \in \mathcal{B}_{f_1}} ||T_1(f_1(t))|| dt + 2 * \int_{t \in \mathcal{B}_{g_1}} ||T_2(g_1(t))|| dt$. Thus:

$$|\Pi_1|_1 \leq |\Pi_2|_1$$

$$\Rightarrow ||T_1|| + ||T_2|| + 2 * \int_{t \in \mathcal{B}_{f_1}} ||T_1(f_1(t))|| dt + 2 * \int_{t \in \mathcal{B}_{g_1}} ||T_2(g_1(t))|| dt \leq ||T_1|| + ||T_2|| + 2 * \int_{t \in \mathcal{B}_{f_2}} ||T_1(f_2(t))|| dt + 2 * \int_{t \in \mathcal{B}_{g_2}} ||T_2(g_2(t))|| dt$$

$$\Rightarrow \int_{t \in \mathcal{B}_{f_1}} ||T_1(f_1(t))|| dt + \int_{t \in \mathcal{B}_{g_1}} ||T_2(g_1(t))|| dt \leq \int_{t \in \mathcal{B}_{f_2}} ||T_1(f_2(t))|| dt + \int_{t \in \mathcal{B}_{g_2}} ||T_2(g_2(t))|| dt$$

$$\Rightarrow |\Pi_1| \leq |\Pi_2|$$

□

THEOREM 2. *Assume we are given two polygonal curves, T_1 and T_2, and a leash length, ε. A non-monotone parameterization of T_1 and T_2 that minimizes the backward movement during the walk can be found in $\mathcal{O}(n^2 \log n)$ time and*

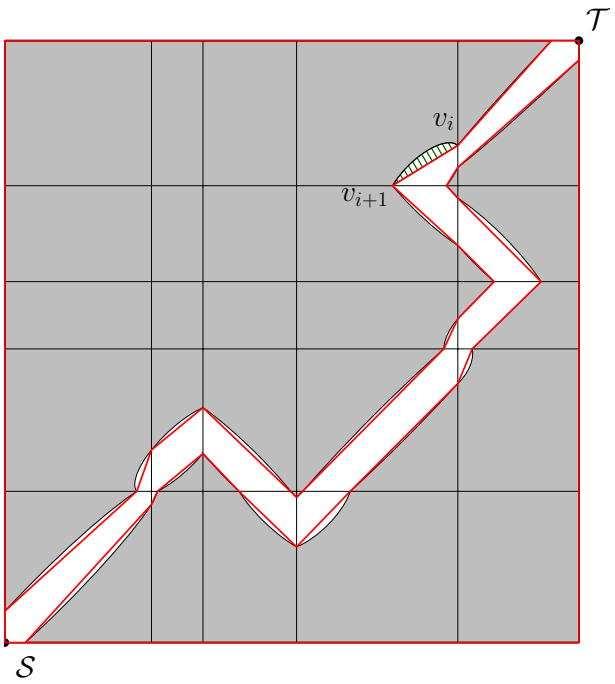

Figure 6: A polygonal domain is constructed by replacing elliptic curves of the boundary of W by line segments.

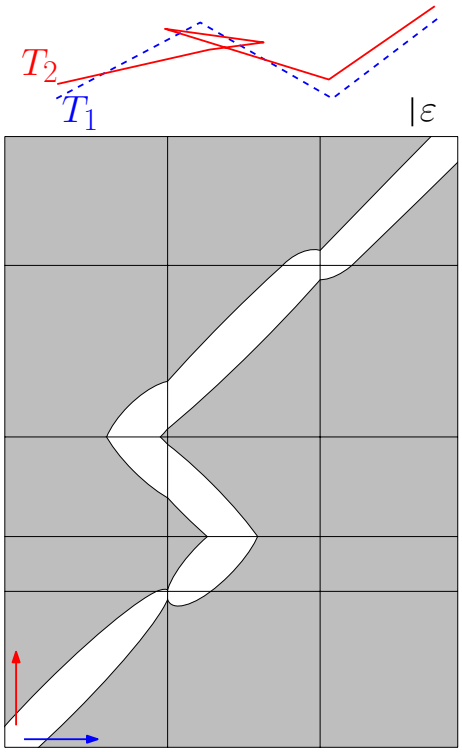

Figure 7: Two polygonal curves T_1 and T_2 and the corresponding deformed free-space diagram are shown. The problem setting allows only backward movement on T_1 (the dashed line).

$\mathcal{O}(n^2)$ space, where n is the number of segments in the input polygonal curves.

PROOF. As we discussed at the beginning of this section, the free-space W is transformed to a polygonal domain $\mathcal{D}$ (Figure 6). Based on Lemma 1, there is an optimal path in $\mathcal{D}$. By Lemma 2, we know that a L_1 shortest path in $\mathcal{D}$ is also an optimal path for the minimum backward Fréchet distance problem. The algorithm by Chen and Wang [6] computes the L_1 shortest path in a polygonal domain, in $\mathcal{O}(k+h\log h)$ time and $\mathcal{O}(k)$ space, where k is the number of vertices of the input polygonal domain and h is the number of holes. In our problem, the size of the polygonal domain $\mathcal{D}$ and also the number of holes is $\mathcal{O}(n^2)$. Therefore, the total time complexity is $\mathcal{O}(n^2\log n)$ and space complexity is $\mathcal{O}(n^2)$. □

The implementation of our proposed algorithm is fairly straightforward. The followings are the main steps of the implementation of this algorithm:

1. The first step is to compute the free-space diagram. This diagram is explained in detail in [1]. Different implementations exist for this purpose. For example, we used the iplete from [7] to produce the figures of this paper. This implementation is open source.

2. The second step is converting the free-space diagram to a polygonal domain. At the beginning of Section 4 we explained how this conversion can be done by processing cells of free-space diagram one by one. The output of this step is a polygonal domain $\mathcal{D}$.

3. The last step is the implementation of the L_1 shortest path algorithm. Chen and Wang's algorithm [6] is theoretically optimal. However, at this time we know of no implementation. There are alternatives for finding the L_1 shortest path in a polygonal domain (e.g., [14]). The easiest one to implement is the visibility graph based algorithm as explained in Section 3.

5. CONCLUSION

We studied a natural optimization problem for the weak Fréchet distance. In this measure, the union of backward movements on the two input curves T_1 and T_2 is minimized, to maintain a weak Fréchet distance of ε. We observed that this problem setting is dual to a weighted shortest path problem in a deformed free-space diagram, $\mathcal{F}$, of T_1 and T_2 in which only the portions of a path from $\mathcal{F}$ are measured if they are not xy-increasing. As a first approach, we showed that a corresponding optimal path in $\mathcal{F}$ is part of the visibility graph between the intersection points of the free-space ellipses with the boundary of the corresponding parameter cells and w.r.t. to the forbidden space of $\mathcal{F}$. This directly led to an algorithm which computes the minimum backward Fréchet distance in polynomial time. Then, as an improvement, we showed that it is not necessary to construct the entire visibility graph and by applying the very recent shortest path algorithm of [6] we obtained an improved running time of $\mathcal{O}(n^2\log n)$ time to solve the minimum backward weak Fréchet problem exactly.

For future work, other versions of the problem could be studied. *1*) The backward movement could be restricted to only one of the input curves (Figure 7). *2*) The cost of backward movement on the input curves could be weighted (it could e.g., be more costly to backtrack on one curve than on the other one.) In this setting, one would want to minimize the weighted sum of all backward movements on the input curves.

6. REFERENCES

[1] H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. *Int. J. Comput. Geometry Appl.*, 5:75–91, 1995.

[2] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. In K. Böhm, C. S. Jensen, L. M. Haas, M. L. Kersten, P.-Å. Larson, and B. C. Ooi, editors, *Proceeding of VLDB*, pages 853–864. ACM, 2005.

[3] K. Buchin, M. Buchin, and Y. Wang. Exact algorithms for partial curve matching via the Fréchet distance. In C. Mathieu, editor, *Proceeding of SODA*, pages 645–654. SIAM, 2009.

[4] M. Buchin, A. Driemel, and B. Speckmann. Computing the Fréchet distance with shortcuts is NP-hard. In *Proceeding of Symposium on Computational Geometry*, SoCG'14, pages 367:367–367:376, New York, NY, USA, 2014. ACM.

[5] J.-L. D. Carufel, A. Gheibi, A. Maheshwari, J.-R. Sack, and C. Scheffer. Similarity of polygonal curves in the presence of outliers. *Computational Geometry: Theory and Applications*, 47(5):625 – 641, 2014.

[6] D. Z. Chen and H. Wang. L_1 Shortest Path Queries among Polygonal Obstacles in the Plane. In *30th International Symposium on Theoretical Aspects of Computer Science (STACS 2013)*, volume 20 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 293–304, 2013.

[7] Free-space diagram ipelet from Günter Rote's homepage. http://www.inf.fu-berlin.de/inst/ag-ti/people/rote/Software/ipelets.html#fsd.

[8] H. Gonzalez, J. Han, X. Li, M. Myslinska, and J. P. Sondag. Adaptive fastest path computation on a road network: A traffic mining approach. In C. Koch, J. Gehrke, M. N. Garofalakis, D. Srivastava, K. Aberer, A. Deshpande, D. Florescu, C. Y. Chan, V. Ganti, C.-C. Kanne, W. Klas, and E. J. Neuhold, editors, *Proceeding of VLDB*, pages 794–805. ACM, 2007.

[9] J. Gudmundsson, P. Laube, and T. Wolle. Movement patterns in spatio-temporal data. In S. Shekhar and H. Xiong, editors, *Encyclopedia of GIS*, pages 726–732. Springer Science and Business Media, 2008.

[10] J. Gudmundsson and N. Valladares. A GPU approach to subtrajectory clustering using the Fréchet distance. In *Proceeding of SIGSPATIAL/GIS*, pages 259–268, 2012.

[11] J. Gudmundsson and T. Wolle. Football analysis using spatio-temporal tools. In *Proceeding of SIGSPATIAL/GIS*, pages 566–569, 2012.

[12] C.-K. Y. Jacob E. Goodman, Janos Pach. Mountain climbing, ladder moving, and the ring-width of a polygon. *The American Mathematical Monthly*, 96(6):494–510, 1989.

[13] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate GPS trajectories. In D. Agrawal, W. G. Aref, C.-T. Lu, M. F. Mokbel, P. Scheuermann, C. Shahabi, and O. Wolfson, editors, *Proceeding of SIGSPATIAL/GIS*, pages 352–361. ACM, 2009.

[14] J. Mitchell. L_1 shortest paths among polygonal obstacles in the plane. *Algorithmica*, 8(1-6):55–88, 1992.