WALKING YOUR DOG IN THE WOODS IN POLYNOMIAL TIME

ERIN WOLF CHAMBERS, ÉRIC COLIN DE VERDIÈRE, JEFF ERICKSON,
SYLVAIN LAZARD, FRANCIS LAZARUS, AND SHRIPAD THITE

ABSTRACT. The Fréchet distance between two curves in the plane is the minimum length of a leash that allows a dog and its owner to walk along their respective curves, from one end to the other, without backtracking. We propose a natural extension of Fréchet distance to more general metric spaces, which requires the leash itself to move continuously over time. For example, for curves in the punctured plane, the leash cannot pass through or jump over the point obstacles ("trees"). We describe a polynomial-time algorithm to compute the *homotopic* Fréchet distance between two given polygonal curves in the plane minus a given set of points.

Date: Last revised July 6, 2007 — ***Preliminary draft, not for public distribution***.

1. Introduction

Given two input curves, a natural question that arises is how similar the curves are to each other. One common measure is the Hausdorff distance, which simply takes the minimum distance between any two points, one from each curve. While the Hausdorff metric does measure closeness in space, it does not take into account the flow of the curves, which in many applications, such as morphing in computer graphics, is an important property of the curves.

The *Fréchet distance*, sometimes called the *dog-leash distance*, is defined as the minimum length of a leash required to connect a dog and its owner as they walk without backtracking along their respective curves from one endpoint to the other. The Fréchet metric takes into account the flow of the two curves because the pairs of points whose distance contributes to the Fréchet metric sweep continuously along their respective curves. It is therefore possible for two curves to have small Hausdorff distance but large Fréchet distance. Fréchet distance is used as a more accurate measure of similarity in many different applications [AG95, AB05].

When the two curves are embedded in a general metric space, the distance between two points on the curves (the length of the shortest leash joining them) is not the Euclidean distance but a geodesic distance. For instance, this is the case if the two curves lie on a terrain (or any surface) [MY05] or if the space containing the two curves has obstacle regions which the leash cannot penetrate [EGHP+02]. The definition of the ordinary Fréchet distance allows the leash to switch discontinuously, without penalty, from one side of an obstacle or a mountain to another.

In this paper, we introduce a continuity requirement on the motion of the leash. We require that the leash cannot switch, discontinuously, from one geodesic to another; in particular, the leash cannot jump over obstacles and can sweep over a mountain only if it is long enough. We define the *homotopic Fréchet distance* between two curves as the Fréchet distance with this additional continuity requirement.

The motion of the leash defines a correspondence between the two curves which can be used to morph between the two curves—two points joined by a leash morph into each other. The homotopic Fréchet distance can thus be thought as the minimal amount of deformation needed to transform one curve into the other.

In robotics, the two curves being compared may be two motion sequences in the configuration space of a robot system. When the configuration space has obstacle regions, the similarity between the two curves is more accurately measured by the homotopic Fréchet distance rather than the ordinary Fréchet distance which ignores obstacles.

Efficiently computing the homotopic Fréchet distance in general metric spaces is a new open problem. We present a polynomial-time algorithm for a special case of the general problem, which is to compute the homotopic Fréchet distance between two polygonal curves in the plane with point obstacles.

2. Definitions

Let S be a fixed Hausdorff metric space. A ***curve*** in S is a continuous function from the unit interval $[0,1]$ to S. We will sometimes abuse notation by using the same symbol to denote a curve $A\colon [0,1] \to S$ and its image in S. A ***reparameterization of*** $\mathbf{[0,1]}$ is a continuous, non-decreasing, onto function $\alpha\colon [0,1] \to [0,1]$. A reparamaterization of a curve $A\colon [0,1] \to S$ is any curve $A \circ \alpha$, where α is a reparamaterization of $[0,1]$. The ***length*** of any curve A, denoted $\mathrm{len}(A)$, is defined by the metric of S; in particular, two reparamaterizations of the same curve have the same length.

A ***leash*** between two curves A and B is another curve $\lambda\colon [0,1] \to S$ such that $\lambda(0) = A(s)$ and $\lambda(1) = B(t)$ for some parameters s and t. A ***homotopy*** between curves A and B is a continuous map $h\colon [0,1] \times [0,1] \to S$ such that $h(\cdot,0) = A$ and $h(\cdot,1) = B$. For any $t \in [0,1]$, the one-parameter function $h(t,\cdot)$ is a leash from A to B. A ***leash map*** between curves A and B is a homotopy between some reparameterization of A and some reparameterization of B. Intuitively, a leash map describes

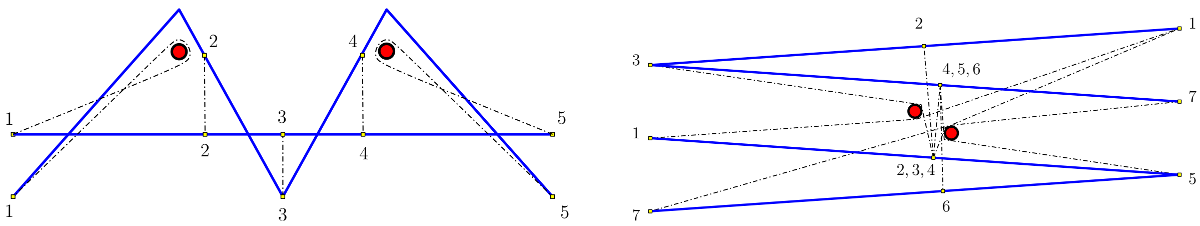

Figure 1. Optimum leash maps for two inputs. Dashed curves between matching numbers represent intermediate leashes.

the *continuous* motion of a leash between a dog walking along A and its owner walking along B. The ***length*** of a leash map ℓ, denoted $\operatorname{len}(\ell)$, is the maximum length of any leash $\ell(t,\cdot)$. Finally, the ***homotopic Fréchet distance*** between two curves A and B, denoted $\overline{\mathcal{F}}(A,B)$, is the infimum, over all leash maps ℓ between A and B, of the length of ℓ:

$$\overline{\mathcal{F}}(A,B) := \inf_{\text{leash map } \ell:[0,1]^2\to S} \left(\max_{0\leqslant t\leqslant 1} \operatorname{len}(\ell(t,\cdot)) \right).$$

In contrast, the classical or ***leashless*** Fréchet distance is defined directly in terms of reparamaterizations and distances:

$$\mathcal{F}(A,B) := \inf_{\alpha,\beta:[0,1]\to[0,1]} \left(\max_{0\leqslant t\leqslant 1} d(A(\alpha(t)), B(\beta(t))) \right).$$

In spaces where shortest paths vary continuously as their endpoints move, such as the Euclidean plane, the two definitions are equivalent. In general, however, homotopic Fréchet distance could be larger (but never smaller) than leashless Fréchet distance.

A ***homotopy relative to A and B***, or simply ***relative homotopy***, is a continuous function $h\colon [0,1]\times[0,1]\to S$, such that $h(\cdot,0)$ and $h(\cdot,1)$ are respectively of the form $A(u(\cdot))$ and $B(v(\cdot))$, where u and v are continuous functions from $[0,1]$ to $[0,1]$.

Two leashes λ and λ' are ***relatively homotopic***, denoted $\lambda \simeq \lambda'$, if there is a relative homotopy h such that $h(0,\cdot)=\lambda$ and $h(1,\cdot)=\lambda'$. It is easy to prove that $\simeq$ is an equivalence relation over the set of leashes. Any leash map is (the transpose of) a relative homotopy; thus, all leashes $\ell(t,\cdot)$ determined by a leash map ℓ lie in the same relative homotopy class.

3. Homotopy in the Punctured Plane

In this paper, we develop a polynomial-time algorithm to compute the homotopic Fréchet distance between two polygonal paths in the ***punctured plane*** $\mathcal{E} = \mathbb{E}^2 \setminus P$, for some finite point set P, where the underlying metric is geodesic distance. The points P act as obstacles; in any leash map in $\mathcal{E}$, the moving leash can neither touch nor jump over any obstacle point.

Specifically, the input to our problem consists of two polygonal curves A and B and a set P of points in the Euclidean plane. Curves A and B may (self-)intersect, but neither curve contains any point in P. To simplify our exposition, we assume that no three vertices of the input (points in P or vertices of A and B) are collinear; this assumption can be enforced algorithmically using standard perturbation techniques [Sei98]. Let $a_0, a_1, \ldots, a_m$ denote the sequence of vertices of A; these points define a standard parameterization $A\colon [0,m]\to\mathcal{E}$ whose restriction to any integer range $[i-1,i]$ is an affine map onto the corresponding edge $a_{i-1}a_i$. Similarly, the vertices $b_0, b_1, \ldots, b_n$ of B define a standard piecewise-affine parameterization $B\colon [0,n]\to\mathcal{E}$. We write $N = n+m+|P|+2$ to denote the total complexity of the input.

Figure 1 illustrates optimum leash maps for a few sample inputs.

3.1. **Geodesics and Geodesic Leash Maps.** To simplify our presentation, we will allow 'paths' in $\mathcal{E}$ to touch obstacles in P. Specifically, we consider ***geodesics***: piecewise-linear curves *in the plane* whose interior vertices are points in P. To ensure that each geodesic lies in a unique homotopy class, we associate a ***turning angle*** with each interior vertex. Let C_ε be a circle centered at obstacle point p and radius ε, small enough to exclude every other point in P. A turning angle of θ at an obstacle point p indicates that replacing the portion of γ inside C_ε with a counterclockwise arc of length $\theta\varepsilon$ around C_ε yields a new path homotopic to γ. For example, a path with turning angle zero makes a U-turn at p without enclosing p; a path that goes straight through p with p on its left (resp. right) has turning angle π (resp. $-\pi$); a turning angle of 10π means the path makes a U-turn after winding around the point five times counterclockwise. A geodesic could meet the same obstacle point more than once; we associate a different turning angle to each incidence.

It can be shown that for any two points x and y in $\mathcal{E}$, the shortest path from x to y in any relative homotopy class is unique and is a geodesic in which every turning angle is either at least π or at most $-\pi$. Conversely, a geodesic in which every turning angle is either at least π or at most $-\pi$ is a shortest path in some homotopy class.

A ***geodesic leash map*** is a leash map $\ell\colon [0,1]\times[0,1]\to\mathbb{E}^2$ in which every leash $\ell(t,\cdot)$ is a geodesic, and all these geodesics are in the same relative homotopy class. We next prove that for any leash map ℓ, there is a shorter geodesic leash map ℓ' in the same homotopy class, with the same parameterizations of A and B.

Lemma 3.1. *Suppose ℓ is a leash map between two curves A and B. There is a geodesic leash map ℓ' between A and B such that, for all $t\in[0,1]$, the leash $\ell'(t,\cdot)$ is the shortest path homotopic to $\ell(t,\cdot)$ with the same endpoints. Additionally, the length of ℓ' is at most the length of ℓ.*

Proof. We lift ℓ to the universal cover $\hat{\mathcal{E}}$ of $\mathcal{E}$, obtaining a leash map $\hat\ell$ between the lifts $\hat A$ and $\hat B$ of A and B respectively. For each $t\in[0,1]$, let $\hat\ell'(\cdot,t)$ be the shortest path between the endpoints of $\hat\ell(t,\cdot)$ which is in the same homotopy class. The universal cover $\hat{\mathcal{E}}$ is a simply-connected space with a locally Euclidean metric, so shortest paths in $\hat{\mathcal{E}}$ vary continuously as the endpoints move continuously. It follows that $\hat\ell'$ is a continuous function in both arguments, and therefore a (geodesic) leash map in $\hat{\mathcal{E}}$. The projection ℓ' of $\hat\ell'$ back to S is a (geodesic) leash map between A and B. For all t, the leash $\ell'(t,\cdot)$ is the shortest path homotopic to $\ell(t,\cdot)$, so $\max_t \operatorname{len}(\ell'(t,\cdot)) \leqslant \max_t \operatorname{len}(\ell(t,\cdot))$. □

This lemma implies that the homotopic Fréchet distance between A and B is the length of a *geodesic* leash map in some homotopy class determined by some reparamaterizations of A and B. Thus, the homotopic Fréchet distance can be redefined as the minimum, over all homotopy classes h, of the *classical* Fréchet distance, where distances are defined by shortest paths in relative homotopy class h:

$$\mathcal{F}_h(A,B) := \min_{\alpha,\beta:[0,1]\to[0,1]} \left(\max_{t\in[0,1]} D_h(A(\alpha(t)), B(\beta(t))) \right)$$

$$\overline{\mathcal{F}}(A,B) := \min_{\text{homotopy class } h} \mathcal{F}_h(A,B)$$

(Here, $D_h(u,v)$ denotes the length of the shortest path from u to v in relative homotopy class h.)

We call a relative homotopy class h ***optimal*** if $\overline{\mathcal{F}}(A,B) = \mathcal{F}_h(A,B)$.

For the rest of the paper, we restrict ourselves to geodesic leashes and geodesic leash maps. In Section 4, we prove that some optimal homotopy class contains a line segment from A to B, and we describe how to enumerate all such homotopy classes in polynomial time. In Section 5, we describe a polynomial-time algorithm to compute the Fréchet distance within a particular homotopy class. Combining these two subroutines gives us a polynomial-time algorithm to compute homotopic Fréchet distance.

3.2. **Maintaining Homotopic Shortest Paths.** Our algorithm relies on observations by Hershberger and Snoeyink [HS94] about shortest homotopic paths in the punctured plane; see also [GS98, CLMS04, EKL06, Bes03, Bes04]. Suppose we already know a shortest path (a leash) λ between points $a \in A$ and $b \in B$, such as, for instance, a straight-line segment ab. To compute the geodesic leash between some other pair of points in the same homotopy class as λ, we follow the continuous evolution of the geodesic as the points a and b move along their respective curves. The sequence of obstacle vertices on the leash behaves like a *double-ended queue* or *deque*. A new vertex is pushed onto one end of the deque whenever the first or last segment of λ collides with an obstacle point. Conversely, a vertex is popped off one end of the deque when the first or last two segments of λ become collinear and the turning angle at their common vertex is either π or $-\pi$.

4. Characterizing an Optimal Homotopy Class

In this section, we prove that there is an optimal homotopy class of leashes that contains a line segment from a point of A and a point of B, and we describe an algorithm to enumerate all homotopy classes with this property in polynomial time.

Let $\text{len}(\lambda)$ denote the length of any geodesic leash λ, and let $\text{turn}(\lambda)$ denote the sum of the absolute values of the turning angles at the interior vertices of λ. (Again, if λ meets the same obstacle point more than once, each incidence separately contributes to $\text{turn}(\lambda)$.) For any pair of leashes λ and λ', we write $\lambda \preceq \lambda'$ if and only if either (1) $\text{len}(\lambda) < \text{len}(\lambda')$, or (2) $\text{len}(\lambda) = \text{len}(\lambda')$ and $\text{turn}(\lambda) \leq \text{turn}(\lambda')$. We write $\lambda \prec \lambda'$ whenever $\lambda \preceq \lambda'$ but $\lambda' \not\preceq \lambda$.

We can extend this relation to homotopy classes as follows. For any relative homotopy class h and any $s, t \in [0, 1]$, let $\sigma_h(s, t)$ denote the shortest path in h between points $A(s)$ and $B(t)$. For any two homotopy classes h and h', we write $h \preceq h'$ if and only if $\sigma_h(s, t) \preceq \sigma_{h'}(s, t)$ for all parameters s and t. We write $h \prec h'$ whenever $h \preceq h'$ but $h' \not\preceq h$.

Lemma 4.1. *For any relative homotopy classes h and h', if $h \preceq h'$, then $\mathcal{F}_h(A, B) \leqslant \mathcal{F}_{h'}(A, B)$.*

Proof. Let ℓ' be an optimum leash map in homotopy class h', so that $\text{len}(\ell') = \mathcal{F}_{h'}(A, B)$. For some reparamaterizations α and β, we have $\ell'(t, \cdot) = \sigma_{h'}(\alpha(t), \beta(t))$ for all t. Let ℓ be the geodesic leash map in homotopy class h defined by the same reparamaterizations: $\ell(t, \cdot) = \sigma_h(\alpha(t), \beta(t))$ for all t. The definition of $\preceq$ implies that $\text{len}(\ell(t, \cdot)) \leqslant \text{len}(\ell'(t, \cdot))$ for all t. It follows that $\mathcal{F}_h(A, B) \leqslant \text{len}(\ell) \leqslant \text{len}(\ell') = \mathcal{F}_{h'}(A, B)$. □

A relative homotopy class h is *minimal* if $h' \preceq h$ implies $h \preceq h'$.

Lemma 4.2. *For any relative homotopy class h, there is a minimal relative homotopy class h' such that $h' \preceq h$.*

Proof. Assume, for the sake of contradiction, that there is no minimal relative homotopy class h' such that $h' \preceq h$. Then, by induction, we can construct an infinite descending chain of relative homotopy classes $h = h_0 \succ h_1 \succ h_2 \succ \cdots$. To simplify notation, let $\sigma_n = \sigma_{h_n}(0, 0)$.

Consider the ordered list of obstacle points on each path σ_n. There are finitely many such ordered lists, because $\text{len}(\sigma_n) \leqslant \text{len}(\sigma_0)$ for each n. Thus, up to taking a subsequence, we may assume that every path σ_n visits the same sequence of obstacle points. This assumption implies that all paths σ_n are geometrically equivalent and thus have equal length. Thus, by definition of $\preceq$, we have $\text{turn}(\sigma_n) < \text{turn}(\sigma_0)$ for all n. There are finitely many relative homotopy classes with a given ordered list of vertices and with bounded total absolute turning angle. (Specifically, since $\text{turn}(\sigma_n) - \text{turn}(\sigma_0)$ is always a multiple of 2π, there are at most $\lfloor \text{turn}(\sigma_0)/2\pi \rfloor + 1$ such classes.) □

The two previous lemmas immediately imply that there is a *minimal* optimal homotopy class. The next lemma characterizes minimal homotopy classes so that we can enumerate them efficiently. A *proper line segment* is a geodesic in E^2 from a point in A to a point in B that does not touch any obstacle point in P.

Lemma 4.3. *A relative homotopy class is minimal if and only if it contains a proper line segment.*

Proof. Let h be the relative homotopy class of a proper line segment σ from $A(s)$ to $B(t)$. For any relative homotopy class $h' \neq h$, the shortest path $\sigma_{h'}(s,t)$ must be longer than σ, so $\sigma_{h'}(s,t) \not\preceq \sigma = \sigma_h(s,t)$, which implies that $h' \not\preceq h$. We conclude that h is minimal.

Let h be an arbitrary minimal homotopy class, and let $\hat{A}$ and $\hat{B}$ be lifts of A and B in the universal cover $\hat{\mathcal{E}}$, such that for all s and t, the shortest path $\hat{\sigma}_h(s,t)$ between $\hat{A}(s)$ and $\hat{B}(t)$ is a lift of $\sigma_h(s,t)$. Let $\hat{P}$ denote the set of all lifts of points in P; these lifted obstacle points lie on the boundary of $\hat{\mathcal{E}}$.

We prove that h contains a proper line segment in two steps. In the first step, we prove that no point $\hat{p} \in \hat{P}$ is a vertex of *every* path $\hat{\sigma}_h(s,t)$. In the second step, we construct a relative homotopy from the initial leash $\sigma_h(0,0)$ to a proper line segment.

Step 1: No common corner. For the sake of deriving a contradiction, suppose there is a lifted obstacle point $\hat{p} \in \hat{P}$ such that for all s and t, the path $\hat{\sigma}_h(s,t)$ has a vertex at $\hat{p}$. For all s and t, the path $\hat{\sigma}_h(s,t)$ is a shortest path, so its turning angle at $\hat{p}$ must lie outside the open interval $(-\pi, \pi)$. This turning angle is a continuous function of s and t, so we can assume without loss of generality that it is always greater than π. In other words, we assume that every path $\hat{\sigma}_h(s,t)$ winds counterclockwise around $\hat{p}$.

Now $\hat{p}$ is a lift of some obstacle $p \in P$, and $\hat{\sigma}_h(s,t)$ similarly projects to a geodesic $\sigma_h(s,t)$. For each s and t, let $\tau(s,t)$ denote the path with the same vertices and turning angles as $\sigma_h(s,t)$, except that the turning angle at p is reduced by 2π. All paths $\tau(s,t)$ belong to a single relative homotopy class, which we denote h'.

Fix parameters s and t, and consider the turning angles of $\sigma_h(s,t)$ and $\tau(s,t)$ at p. If the turning angle of $\sigma_h(s,t)$ at p is strictly between π and 3π, then the turning angle of $\tau(s,t)$ at p is strictly between $-\pi$ and π. In this case, $\tau(s,t)$ cannot be the shortest path from s to t in this homotopy class, so $\operatorname{len}(\sigma_{h'}(s,t)) < \operatorname{len}(\tau(s,t)) = \operatorname{len}(\sigma_h(s,t))$.

On the other hand, if the turning angle of $\sigma_h(s,t)$ at p is at least 3π, then the turning angle of $\tau(s,t)$ at p is at least π, which implies that $\tau(s,t)$ is the shortest path from s to t in h'. In this case $\sigma_h(s,t)$ and $\sigma_{h'}(s,t) = \tau(s,t)$ are geometrically equivalent and thus have equal length, but $\operatorname{turn}(\sigma_{h'}(s,t)) = \operatorname{turn}(\sigma_h(s,t)) - 2\pi < \operatorname{turn}(\sigma_h(s,t))$.

Hence $\sigma_{h'}(s,t) \prec \sigma_h(s,t)$ for all s and t, which contradicts our assumption that h is a minimal homotopy class. We conclude that no point $\hat{p}$ lies on every shortest path $\hat{\sigma}_h(s,t)$.

Step 2: Homotopy construction. If the shortest path $\hat{\sigma}_h(0,0)$ is a line segment, then the geodesic $\sigma_h(0,0)$ is also a line segment, and the proof is complete. Thus, we assume that $\hat{\sigma}_h(0,0)$ has at least one interior vertex in $\hat{P}$.

Let $\hat{p}_1, \ldots, \hat{p}_k$ be the sequence of lifted obstacle points on the shortest path $\hat{\sigma}_h(0,0)$. (The points $\hat{p}_i$ are distinct, although their projections back into $\mathcal{E}$ might not be.) Our previous argument implies that for each i, there is a pair of parameters (s_i, t_i) such that $\hat{\sigma}_h(s_i, t_i)$ does not pass through $\hat{p}_i$.

We consider a continuous motion of the parameter point (s,t), starting at $(s,t) = (0,0)$ and then moving successively to each point (s_i, t_i). Specifically, we define two continuous functions $s\colon [0,k] \to [0,m]$ and $t\colon [0,k] \to [0,n]$ such that $s(0) = t(0) = 0$, and for any integer i, we have $s(i) = s_i$ and $t(i) = t_i$. To simplify our notation, we write $\hat{\sigma}(\tau)$ to denote the shortest path $\hat{\sigma}_h(s(\tau), t(\tau))$.

As the parameter τ ('time') increases, points in $\hat{P}$ are inserted into and deleted from the deque of vertices of $\hat{\sigma}(\tau)$. If the deque is empty at any time τ, then the shortest path $\hat{\sigma}(\tau)$ is a proper line segment, which implies that the projected path $\sigma(\tau)$ is a proper line segment as well, concluding the proof. Thus, we assume to the contrary that the deque is never empty. Each vertex $\hat{p}_1, \ldots, \hat{p}_k$ must be deleted from the deque at some time during the motion (but may be reinserted later).

Suppose $\hat{p}$ is the *last* point among $\hat{p}_1, \ldots, \hat{p}_k$ to be removed from the deque for the *first* time. Without loss of generality, we assume $\hat{p}$ is first removed from the front of the deque at time τ_1. Let $\hat{q}$ denote the second point in the deque just before $\hat{p}$ is removed; this point must exist, because the deque is never empty. The point $\hat{p}$ lies on the first segment $\hat{a}\hat{q}$ of $\hat{\sigma}(\tau_1)$, where $\hat{a} = \hat{A}(s(\tau_1))$.

By definition of $\hat{p}$, point $\hat{q}$ must have been pushed onto the *back* of in the deque at some earlier time $\tau_2 < \tau_1$. Just after $\hat{q}$ is inserted, the last two points in the deque are $\hat{p}$ and $\hat{q}$, in that order. Moreover, $\hat{q}$ lies on the last segment $\hat{p}\hat{b}$ of $\hat{\sigma}(\tau_2)$, where $\hat{b} = \hat{B}(t(\tau_2))$.

Thus, there is an *improper* line segment $\hat{a}\hat{b}$ between a point in $\hat{A}$ and a point in $\hat{B}$. Since all line segments in $\hat{\mathcal{E}}$ are shortest paths, $\hat{a}\hat{b}$ is the shortest path $\hat{\sigma}_h(\tau_1, \tau_2)$. Thus, the path $\sigma_h(\tau_1, \tau_2)$ in $\mathcal{E}$ is an *improper* line segment in relative homotopy class h. Finally, for sufficiently small $\varepsilon > 0$, one of the four paths $\sigma_h(\tau_1 \pm \varepsilon, \tau_2 \pm \varepsilon)$ is a proper line segment (because no three vertices of the input are collinear). □

We have proved our main proposition.

Proposition 4.4. *Some optimal relative homotopy class contains a line segment from A to B.*

We can enumerate the set of minimal relative homotopy classes in polynomial time as follows. For every pair of points $p, q \in P$, we find all intersections of the line $\overleftrightarrow{pq}$ with A and B, in $O(m+n)$ time by brute force. For each pair of intersection points $a \in A$ and $b \in B$, we obtain four proper line segments arbitrarily close to ab. (Alternately, we obtain four different straight geodesics from a to b, by assigning turning angles π or $-\pi$ at p and q.) Altogether, we find $O(mn|P|^2)$ proper line segments, at least one in each minimal homotopy class, in $O(mn|P|^2)$ time.

There are polygonal curves and point sets that admit $\Omega(mn|P|^2)$ distinct minimal relative homotopy classes; see Figure 2 for an example. Thus, any improvement in this portion of the algorithm will require a finer characterization of optimal relative homotopy classes.

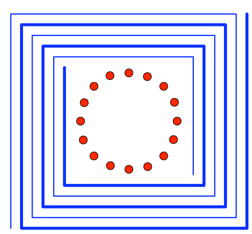

FIGURE 2. Curves and points with $\Omega(N^4)$ minimal relative homotopy classes.

5. Computing Fréchet Distance in One Homotopy Class

In this section, we describe an algorithm to compute the Fréchet distance $\mathcal{F}_h(A, B)$ in some relative homotopy class h. Our algorithm is a direct adaptation of Alt and Godau's algorithm for computing the classical Fréchet distance between polygonal paths in the plane [AG95].

As in the previous section, for any $s \in [0, m]$ and $t \in [0, n]$, let $\sigma_h(s, t)$ denote the shortest path from $A(s)$ to $B(t)$ in homotopy class h, and let $d_h(s, t) = \text{len}(\sigma_h(s, t))$. For any $\varepsilon > 0$, let $F_\varepsilon \subseteq [0, m] \times [0, n]$ denote the ***free space*** $\{(s, t) \mid d_h(s, t) \leqslant \varepsilon\}$. Our goal is to compute the smallest value of ε such that F_ε contains a monotone path from $(0, 0)$ to (m, n); this is precisely the Fréchet distance $\mathcal{F}_h(A, B)$.

The parameter space $[0, m] \times [0, n]$ decomposes naturally into an $m \times n$ grid; let $C_{i,j} = [i-1, i] \times [j-1, j]$ denote the grid cell representing paths from the ith edge of A to the jth edge of B. Our generalization of Alt and Godau's algorithm requires that the restriction of the function d_h to any grid cell $C_{i,j}$ is convex. We prove this fact in Appendix A (Proposition A.5).

As input to our problem, we are given a path $\sigma_h(s_0, t_0)$ in relative homotopy class h; by the results in the previous section, we can assume that $\sigma_h(s_0, t_0)$ is a proper line segment. Without loss of generality, we assume that the endpoints $A(s_0)$ and $B(t_0)$ are vertices of A and B; otherwise, we insert them as new vertices and reparameterize.

5.1. **Preprocessing for Distance Queries.** The only significant difference between our algorithm and Alt and Godau's is that we require additional preprocessing to compute several *critical distances* and an auxiliary data structure to answer certain *distance queries.* (If there are no obstacles, each critical distance can be computed, and each distance query can be can be answered, in constant time.) There are three types of critical distances:

- ***endpoint distances*** $d_h(0, 0)$ and $d_h(m, n)$,
- ***vertex-edge distances*** $d_h(i, [j-1, j]) = \min\{d_h(i, t) \mid t \in [j-1, j]\}$ for all integers $i \in [0, m]$ and $j \in [1, n]$, and
- ***edge-vertex distances*** $d_h([i-1, i], j) = \min\{d_h(s, j) \mid s \in [i-1, i]\}$ for all integers $i \in [1, m]$ and $j \in [0, n]$.

Given integers i and j and any real value ε, a ***horizontal distance query*** asks for all values of $t \in [j-1, j]$ such that $d_h(i, t) = \varepsilon$, and a ***vertical distance query*** asks for all values of $s \in [i-1, i]$ such that $d_h(s, j) = \varepsilon$. The convexity of d_h within any grid cell implies that any distance query returns at most two values.

We first describe how to preprocess a single vertical edge in the parameter grid; obviously a similar result applies to horizontal grid edges.

Lemma 5.1. *Suppose we are given the path $\sigma_h(i, j)$ and its length $d_h(i, j)$, for some integers $i \in [0, m]$ and $j \in [1, n]$. In $O(|P| \log|P|)$ time, we can build a data structure of size $O(|P|)$, such that for any ε, all values of $t \in [j-1, j]$ such that $d_h(i, t) = \varepsilon$ can be computed in $O(\log|P|)$ time. In the same time, we also compute the critical vertex-edge distance $d_h(i, [j-1, j])$, the path $\sigma_h(i, j-1)$, and its length $d_h(i, j-1)$.*

Proof. We first compute a triangulation of the points $P \cup \{a_i, b_{j-1}, b_j\}$ that includes the segment $b_{j-1}b_j$ as an edge (for example, a constrained Delaunay triangulation) in time $O(|P| \log|P|)$.

We will need the following observations used in the *funnel* algorithm for computing shortest homotopic paths [Cha82, LP84, HS94]. The shortest homotopic paths $\sigma_h(i, j)$ and $\sigma_h(i, j-1)$ may travel together for some time, and then split at some vertex v; the remaining parts of these paths are concave chains that form a funnel with apex v. The funnel is embedded in the plane, which implies that each of both concave chains intersects a given edge of the triangulation at most twice and touches every obstacle at most once (in particular, it has complexity at most $|P|$).

We walk on $\sigma_h(i, j)$ back from b_j to a_i, and we stop at the first vertex w (considering turning angles for the concavity) such that $\sigma_h(i, j)$ is non-concave at w or such that the (concave) chain from w to b_j crosses an edge of the triangulation more than twice. In particular, we walk on $O(|P|)$ edges of the path. Let us denote by $\sigma_h(i, j)_1$ and $\sigma_h(i, j)_2$ the subpaths of $\sigma_h(i, j)$ before and after w, respectively. By the observations above, $\sigma_h(i, j)_2$ contains the concave chain of the funnel from v to b_j. In particular, $\sigma_h(i, j-1)$ is the concatenation of $\sigma_h(i, j-1)_1 = \sigma_h(i, j)_1$ and of $\sigma_h(i, j-1)_2$, the shortest path from w to b_{j-1} that travels through the same triangles as $\sigma_h(i, j)_2$. Since $\sigma_h(i, j)_2$ enters $O(|P|)$ triangles by the choice of w, we can compute $\sigma_h(i, j-1)_2$ in $O(|P|)$ time using the funnel algorithm. We then compute the apex v and the concave chains of the funnel.

The paths $\sigma_h(i, t)$, for t between $j-1$ and j, all begin with the initial part σ of $\sigma_h(i, j)$ from a_i to v. Path σ has unknown complexity, but its length is the difference between the length of $\sigma_h(i, j)$ and the length of the chain of the funnel from v to b_j, so we can compute it in $O(|P|)$ time. Now, using the funnel algorithm, we kinetically maintain the part of $\sigma_h(i, t)$ after σ and its length, as t decreases from j to $j-1$. During the motion, each point in P is pushed or popped at most once, so the number of events is $O(|P|)$ and the time spent is $O(|P|)$.

Between events, the function $d_h(i,t)$ has the form $\|p - B(t)\| + L$, where p is the second-to-last vertex of $\sigma_h(i,t)$ and L is the (constant) length of the subpath of $\sigma_h(i,t)$ from a_i to p. Let t^* be the coordinate defining the critical vertex-edge distance: $d_h(i,[j-1,j]) = d_h(i,t^*)$. By convexity of d_h within each grid cell, t^* is unique; it is either one of the endpoints $j-1$ or j, or one of the local minima of the function $\|p - B(t)\|$ between events. To represent the graph of distances, we record the sequence of event times t (including t^*), the distances $d_h(i,t)$ at each event t, and the pair (L,p) for each interval between events.

Finally, in $O(|P|)$ time, we replace the final part of $\sigma_h(i,j)$, which is one concave chain of the funnel, with the other concave chain of the funnel. This takes $O(|P|)$ time, and the result is $\sigma_h(i,j-1)$.

After the preprocessing ends, we answer distance queries in $O(\log|P|)$ time as follows. If $\varepsilon < d_h(i,[j-1,j])$, we return the empty set. If $\varepsilon = d_h(i,[j-1,j])$, we return only t^*. If $\varepsilon > d_h(i,t^*)$, we perform two binary searches, one over the events between $j-1$ and t^*, the other over the events between t^* and j, to find time intervals containing the desired parameter values. For each interval found, we retrieve the corresponding pair (L,p) and return the unique value t such that $\|p - B(t)\| + L = \varepsilon$. □

Lemma 5.2. *In $O(mn|P|\log|P|)$ time, we can compute all critical distances, as well as a data structure of size $O(mn|P|)$ that can answer any horizontal or vertical distance query in $O(\log|P|)$ time.*

Proof. The idea is to preprocess each edge of the parameter grid as in Lemma 5.1. We start from the vertex (i,j) of the grid for which we know that $\sigma_h(i,j)$ is a straight-line segment. We then walk on the edges of the grid, visiting each edge at least once and at most $O(1)$ times. During this walk, at each current vertex (i,j), we maintain the shortest homotopic path $\sigma_h(i,j)$ and its length $d_h(i,j)$. Each time we walk along an edge, we apply Lemma 5.1 to preprocess it and to compute the shortest homotopic path corresponding to the target vertex of that edge. Each step takes $O(|P|\log|P|)$ time, and there are $O(mn)$ edges, whence the running-time. (We can't afford copying shortest homotopic paths $\sigma_h(i,j)$, because a constant fraction of them may have complexity $\Omega(|P|(m+n))$.) □

5.2. **Decision Procedure.** Like Alt and Godau, we first consider the following *decision problem*: Is $\mathcal{F}_h(A,B)$ at least some given value ε? Equivalently, is there a monotone path in the free space F_ε from $(0,0)$ to (m,n)? Our algorithm to solve this decision problem is identical to Alt and Godau's, except for the $O(\log|P|)$-factor penalty for distance queries.

Fix ε. We define the *reachability space* R_ε as the set of parameter points (s,t) such that there is a monotone path in F_ε from $(0,0)$ to (s,t). Our goal is to determine whether R_ε contains the point (m,n).

For any integers i and j, let $h_{i,j}$ denote the intersection of the free space F_ε with the horizontal edge $([i-1,i],j)$, and let $v_{i,j}$ denote the intersection of F_ε with the vertical edge $(i,[j-1,j])$. In the first phase of the decision procedure, we compute $h_{i,j}$ and $v_{i,j}$ for all i and j, using one distance query for each edge of the parameter grid.

Similarly, let $\hbar_{i,j}$ denote the intersection of R_ε with the horizontal edge $([i-1,i],j)$, and let $\overline{v}_{i,j}$ denote the intersection of R_ε with the vertical edge $(i,[j-1,j])$. In the second phase of the decision procedure, we compute these segments in $O(mn)$ time by visiting the cells in lexicographic order, from $C_{1,1}$ to $C_{m,n}$. Specifically, given segments $\hbar_{i,j}$, $\overline{v}_{i,j}$, $h_{i,j+1}$, and $v_{i+1,j}$ on the boundary of $C_{i,j}$, we can compute $\hbar_{i,j+1}$ and $\overline{v}_{i+1,j}$ in constant time.

Finally, our decision algorithm returns TRUE if and only if $(m,n) \in \hbar_{m,n}$. The total running time of our decision procedure is $O(mn\log|P|)$.

5.3. **Computing Fréchet Distance.** Finally, we describe how to use our decision procedure to compute the optimum value $\varepsilon^* = \min\{\varepsilon \mid (m,n) \in R_\varepsilon\}$; this is the Fréchet distance $\mathcal{F}_h(A,B)$.

Again, our strategy is almost identical to Alt and Godau's, but with some additional complications due to distance queries.

We start by computing the critical distances and the distance-query data structure in time $O(mn|P|\log|P|)$, as described in Lemma 5.2. We then sort the $O(mn)$ critical distances. Using the decision procedure, we can compare the optimal distance ε^* with any critical distance ε in $O(mn\log|P|)$ time. By binary search, we can, repeating this step $O(\log mn)$ times, compute an interval $[\varepsilon^-, \varepsilon^+]$ that contains ε^* but no critical distances.

Finally, we apply Megiddo's *parametric search* technique [Meg83]; see also [vOV04]. Parametric search combines our decision procedure with a 'generic' parallel algorithm whose combinatorial behavior changes at the optimum value ε^*. Alt and Godau observe that one of two events occurs when $\varepsilon = \varepsilon^*$:

- For some integers i, i', j, the bottom endpoint of $v_{i,j}$ and the top endpoint of $v_{i',j}$ lie on the same horizontal line.
- For some integers i, j, j', the left endpoint of $h_{i,j}$ and the right endpoint of $h_{i,j'}$ lie on the same vertical line.

Thus, it suffices to use a 'generic' algorithm that sorts the $O(mn)$ endpoint *values* of all non-empty segments $h_{i,j}$ and $v_{i,j}$. The value of an endpoint (s, j) of $h_{i,j}$ is s. Similarly, the value of an endpoint (i, t) of $v_{i,j}$ is t.

As suggested by Cole [Col87], we use the parallel sorting algorithm of Ajtai et al. [AKS83] ('the AKS network'). Each parallel step of the sorting algorithm needs to compare $O(mn)$ endpoints. The graph of an endpoint, considered as a function of ε, is convex, monotone, and made of $O(|P|)$ pieces, each having a simple closed form (see proof of Lemma 5.1). It follows that the sign of a comparison between two endpoints may change at $O(|P|)$ different values of ε that can be computed in $O(|P|)$ time. Applying the parametric search paradigm requires the following operations for each parallel step of the sorting algorithm:

- Compute the $O(mn|P|)$ values of ε corresponding to the changes of sign of the $O(mn)$ comparisons. This can be done in $O(mn|P|)$ time.
- Apply binary search to these values by median finding, calling the decision procedure to discard half of them at each step of the search. This takes $O(mn|P| + T_d \log(mn|P|))$ time, where the first term in this sum stands for the computation of the medians during binary search and $T_d = O(mn\log|P|)$ is the complexity of the decision procedure. We obtain this way an interval for ε where each of the $O(mn)$ comparisons has a determined sign.
- Deduce in $O(mn\log|P|)$ time the sign of each of the $O(mn)$ comparisons in the previously computed interval.

Taking into account the $O(\log mn)$ parallel steps of the sorting algorithm, the resulting parametric search algorithm runs in time $O(mn\log(mn)(|P| + \log|P|\log(mn|P|))$.

Lemma 5.3. *Given a proper line segment in relative homotopy class h, the Fréchet distance $\mathcal{F}_h(A, B)$ can be computed in time $O(mn|P|\log(mn|P|)) = O(N^3\log N)$.*

5.4. **Summary.** Finally, to compute the homotopic Fréchet distance $\overline{\mathcal{F}}(A, B)$, we compute the Fréchet distance $\mathcal{F}_h(A, B)$ in each of the $O(mn|P|^2)$ homotopy classes h that contain a line segment. We conclude:

Theorem 5.4. *The homotopic Fréchet distance between two polygonal curves in the punctured plane can be computed in $O(m^2n^2|P|^3\log(mn|P|)) = O(N^7\log N)$ time.*

6. Conclusion

In this paper, we introduced a natural generalization of the Fréchet distance between curves to more general metric spaces, called the homotopic Fréchet distance. We gave a polynomial-time algorithm to compute the homotopic Fréchet distance between polygonal curves in the punctured plane.

Improving the running time of our algorithm is the most immediate outstanding open problem. We conjecture that the running time can be improved significantly by optimizing leash maps in every minimal homotopy class simultaneously to compute the leash map of overall minimum length. Since shortest paths between the same endpoints but belonging to different homotopy classes are related, we expect to (partially) reuse the results of shortest path computations going from one homotopy class to another.

The *weak* Fréchet distance is a variant of the ordinary dog-leash distance without the requirement that the endpoints move monotonically along their respective curves—the dog and its owner are allowed to backtrack to keep the leash between them short. Alt and Godau [AG95] gave a simpler algorithm for computing the weak Fréchet distance, using a graph shortest path algorithm instead of parametric search. A similar simple algorithm computes the weak (non-monotone) version of homotopic Fréchet distance in $O(N^7 \log N)$ time. We omit further details from this extended abstract.

It would be interesting to compute optimum leash maps in more general spaces. In particular, we are interested in computing the homotopic Fréchet distance between two curves on a convex polyhedron, generalizing the algorithm of Maheshwari and Yi for leashless Fréchet distance [MY05]. The vertices of the polyhedron are 'mountains' over which the leash can pass only if it is long enough. Shortest paths on the surface of a convex polyhedron do *not* vary continuously as the endpoints move, because of the positive curvature at the vertices, so we cannot consider only geodesic leash maps.

We expect that homotopic Fréchet distance is more useful in several if not all applications that use the ordinary Fréchet distance. For instance, two paths in the configuration space of a robot system are more accurately compared by the homotopic Fréchet distance between them when the configuration space has obstacle regions. It is therefore an interesting open problem to extend our algorithm to curves in the plane with more general obstacles. A significant challenge is to generalize our characterization of the optimum homotopy class from Section 4 to the case of arbitrary obstacle regions.

It would also be interesting to consider the homotopic Fréchet distance between higher-dimensional manifolds, because such problems arise with respect to surfaces in configuration spaces of robot systems; however, even the ordinary Fréchet distance is difficult to compute in higher dimensions [AB05].

Acknowledgments

This research was initiated during a visit to INRIA Lorraine in Nancy, made possible by a UIUC/CNRS/INRIA travel grant. Research by Erin Chambers and Jeff Erickson was also partially supported by NSF grant DMS-0528086. Research by Shripad Thite was partially supported by the Netherlands Organisation for Scientific Research (NWO) under project number 639.023.301 and by INRIA Lorraine. We thank Hazel Everett and Sylvain Petitjean for useful discussions and great company, and Kira for several walks in the woods.

References

[AB05] H. Alt and M. Buchin. Semi-computability of the Fréchet distance between surfaces. In *Proc. 21st European Workshop on Computational Geometry*, pages 45–48, March 2005.

[AG95] H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. *IJCGA*, 5(1–2):75–91, 1995.

[AKS83] M. Ajtai, J. Komlós, and E. Szemerédi. Sorting in $c \log n$ parallel steps. *Combinatorica*, 3(1):1–19, 1983.

[Bes03] S. Bespamyatnikh. Computing homotopic shortest paths in the plane. In *Proc. 14th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 609–617, 2003.

[Bes04] S. Bespamyatnikh. Encoding homotopy of paths in the plane. In *Proc. LATIN 2004: Theoretical Infomatics*, volume 2976 of *Lect. Notes Comput. Sci.*, pages 329–338. Springer-Verlag, 2004.

[Cha82] B. Chazelle. A theorem on polygon cutting with applications. In *Proc. 23rd Annu. IEEE Sympos. Found. Comput. Sci.*, pages 339–349, 1982.

[CLMS04] S. Cabello, Y. Liu, A. Mantler, and J. Snoeyink. Testing homotopy for paths in the plane. *Discrete Comput. Geom.*, 31(1):61–81, 2004.

[Col87] R. Cole. Slowing down sorting networks to obtain faster sorting algorithms. *JACM*, 34(1):200–208, January 1987.

[EGHP$^+$02] A. Efrat, L. J. Guibas, S. Har-Peled, J. S. B. Mitchell, and T. M. Murali. New similarity measures between polylines with applications to morphing and polygon sweeping. *Discrete Comput. Geom.*, 28:535–569, 2002.

[EKL06] A. Efrat, S. G. Kobourov, and A. Lubiw. Computing homotopic shortest paths efficiently. *Comput. Geom. Theory Appl.*, 35(3):162–172, 2006.

[GS98] D. Grigoriev and A. Slissenko. Polytime algorithm for the shortest path in a homotopy class amidst semi-algebraic obstacles in the plane. In *Proc. Internat. Sympos. Symbolic and Algebraic Computation*, pages 17–24, 1998.

[HS94] J. Hershberger and J. Snoeyink. Computing minimum length paths of a given homotopy class. *Computational Geometry: Theory and Applications*, 4:63–67, 1994.

[LP84] D. T. Lee and F. P. Preparata. Euclidean shortest paths in the presence of rectilinear barriers. *Networks*, 14:393–410, 1984.

[Meg83] N. Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *J. ACM*, 30:852–866, 1983.

[MY05] A. Maheshwari and J. Yi. On computing Fréchet distance of two paths on a convex polyhedron. In *Proc. 21st European Workshop on Computational Geometry*, pages 41–44, 2005.

[Sei98] R. Seidel. The nature and meaning of perturbations in geometric computing. *Discrete Comput. Geom.*, 19:1–17, 1998.

[vOV04] R. van Oostrum and R. C. Veltkamp. Parametric search made practical. *Computational Geometry: Theory and Applications*, 28:75–88, 2004.

Appendix A. Convexity of Geodesic Distance

Let A and B be arbitrary line segments in the universal cover $\hat{\mathcal{E}}$ of the punctured plane, and let $d\colon [0,1] \to \mathbb{R}^+$ be such that $d(t)$ is the length of the shortest path in $\hat{\mathcal{E}}$ between $\hat{A}(t)$ and $\hat{B}(t)$. The goal of this section is to prove that this distance function is convex.

We first give some elementary lemmas concerning convex functions. Let $f : \mathbb{R} \to \mathbb{R}^2$ be an arbitrary affine function. Let $\|\cdot\|$ denote the Euclidean norm.

Lemma A.1. *The function $t \mapsto \|f(t)\|$ is convex.*

Proof. For any $s, t \in \mathbb{R}$ and for any $\alpha \in [0,1]$, we have

$$\|f((1-\alpha)s + \alpha t)\| = \|(1-\alpha)f(s) + \alpha f(t)\|$$

From the triangle inequality and the homogeneity of the norm, we get

$$\|f((1-\alpha)s + \alpha t)\| \leqslant (1-\alpha)\|f(s)\| + \alpha\|f(t)\|$$

which proves convexity. □

Lemma A.2. *Let f and g be two real convex functions such that $f(t) \geqslant g(t)$ for all $t \in \mathbb{R}$, and let $\tau \in \mathbb{R}$ be such that $f(\tau) = g(\tau)$. Then the two functions $u(t)$ and $v(t)$ defined below are convex:*

$$u(t) = \begin{cases} f(t), & \text{if } t < \tau \\ g(t), & \text{otherwise} \end{cases} \qquad v(t) = \begin{cases} g(t), & \text{if } t < \tau \\ f(t), & \text{otherwise} \end{cases}$$

Proof. We prove the convexity of u; the convexity of v can be proved similarly. We need to show:

$$\forall s, t \text{ such that } s < \tau < t, \forall \alpha \in [0,1] : u((1-\alpha)s + \alpha t) \leqslant (1-\alpha)u(s) + \alpha u(t) \tag{1}$$

Let $t = (1-\alpha)s + \alpha t$. If $t \geqslant \tau$ then $u(t) = g(t)$, and the inequality follows from the convexity of g and because $g(t) \leqslant u(t)$.

If $t < \tau$ Equation 1 becomes

$$f(t) \leqslant f(s) + \alpha(g(t) - f(s))$$

Let

$$\rho_g(x,y) = \frac{g(y) - g(x)}{y - x}$$

Since g is convex, it is well-known that ρ_g is an increasing function of each variable when the other one is fixed.

Let $t = s + \beta(\tau - s)$. By convexity of f, we have

$$f(t) \leqslant f(s) + \beta(f(\tau) - f(s))$$

whence we conclude

$$\begin{aligned} f(t) &\leqslant f(s) + \beta(g(\tau) - g(s)) - \beta(f(s) - g(s)) \\ &\leqslant f(s) + \beta(g(\tau) - g(s)) - \alpha(f(s) - g(s)) \end{aligned} \tag{2}$$

The last inequality is implied by $\alpha < \beta$. Since $\beta(\tau - s) = \alpha(t - s)$ we can also write

$$\begin{aligned} \beta(g(\tau) - g(s)) &= \beta(\tau - s)\rho_g(\tau, s) \\ &\leqslant \alpha(t - s)\rho_g(t, s) \end{aligned} \tag{3}$$

Substituting Equation 3 in Equation 2, we get

$$f(t) \leqslant f(s) + \alpha(t - s)\rho_g(t, s) - \alpha(f(s) - g(s))$$

which is the required inequality after expanding the right-hand side. □

Corollary A.3. *Let f, g, h be three real convex functions such that $f(t) \geq h(t)$ and $g(t) \geq h(t)$ for all $t \in \mathbb{R}$, and let $\tau \in \mathbb{R}$ be such that $f(\tau) = g(\tau) = h(\tau)$. Then the function w defined below is convex:*

$$w(t) = \begin{cases} f(t), & \text{if } t < \tau \\ g(t), & \text{otherwise} \end{cases}$$

Proof. Define

$$u(t) = \begin{cases} f(t), & \text{if } t < \tau \\ h(t), & \text{otherwise} \end{cases} \qquad v(t) = \begin{cases} h(t), & \text{if } t < \tau \\ g(t), & \text{otherwise} \end{cases}$$

Then, we have $w(t) = \max\{u(t), v(t)\}$ and the convexity of w follows from the convexity of u and v by Lemma A.2. □

A function real f is *locally convex* if every $t \in \mathbb{R}$ has an open neighborhood $N(t)$ such that the restriction of f to $N(t)$ is convex.

Lemma A.4. *Every locally convex function is convex.*

Proof. Fix $r, t \in \mathbb{R}$ and $\alpha \in [0, 1]$. Consider a finite family of intervals covering $[r, t]$ such that f is convex over each of the intervals. Such a family exists by compactness of $[r, t]$. Let 2ε be a Lebesgues number for this covering: for all $s \in [r, t]$ the interval $[s - \varepsilon, s + \varepsilon]$ is contained in one of the intervals of the covering. Consider a sequence $r = s_1 < s_2 < \ldots < s_n = t$ such that $s_{i+1} - s_i \leq \varepsilon$ for all $i \in [1, n-1]$. Let g be the piecewise-linear function over $[s, t]$ defined by $g(s_i) = f(s_i)$ for all $i \in [1, n]$. The graph of g is convex and above the graph of f. It follows that $f((1-\alpha)r + \alpha t) \leq g((1-\alpha)r + \alpha t) \leq (1-\alpha)f(r) + \alpha f(t)$. □

Proposition A.5. *The function d is convex.*

Proof. Let $\sigma(t)$ be the shortest path from $A(t)$ to $B(t)$, and let $\hat{P}$ denote the set of lifts of obstacles points in P to the universal cover $\hat{\mathcal{E}}$. Consider the sequence $V(t)$ of points in $\hat{P}$ where $\sigma(t)$ bends. As t increases from 0 to 1, the sequence $V(t)$ evolves as follows.

(1) If $V(t)$ is empty, then $\sigma(t)$ is a line segment and $V(t)$ remains empty as t increases until $\sigma(t)$ hits a point in $\hat{P}$ (or possibly two points simultaneously). Just after this event, $V(t)$ contains that point (or points).
(2) If $V(t)$ is non-empty, then $\sigma(t)$ comprises a first segment $r(t)$ pivoting around the first vertex in $V(t)$ and a last segment $s(t)$ pivoting around the last vertex in $V(t)$. The rest of $\sigma(t)$ remains fixed until one of the following events occurs:
   - either $r(t)$ or $s(t)$ hits a vertex of T;
   - $r(t)$ becomes aligned with the next segment along $\sigma(t)$; or
   - $s(t)$ becomes aligned with the previous segment along $\sigma(t)$.

   Each of these events changes the sequence $V(t)$ by adding or removing at most two vertices; at all other times, $V(t)$ is constant.

The first type of event can happen only once during the motion. The same can be said of the reverse event when $r(t)$ becomes aligned with $s(t)$ and $V(t)$ becomes empty. All other events correspond to $r(t)$ or $s(t)$ becoming aligned with a segment defined by two points in $\hat{P}$. For example, if $r(t)$ hits $\hat{p} \in \hat{P}$, then $r(t)$ is aligned with $\hat{p}$ and the first vertex in $V(t)$. It follows that the number of events is finite.

Between two consecutive events, the length of $\sigma(t)$ is a sum of functions that are either constant or of the type $\|A(t) - B(t)\|$, $\|A(t) - v\|$, or $\|B(t) - w\|$ for some pivoting vertices v and w. Each of these functions is convex by Lemma A.1, hence d is convex between two consecutive events. Due to Lemma A.4 it suffices to prove that d is locally convex at each event time to prove that it is convex over the whole unit interval.

Consider an event time τ. If $V(t)$ contains the same vertex $\hat{p}$ just before and just after τ, then we can split $\sigma(t)$ into two parts $q(t)$ from $A(t)$ to $\hat{p}$ and $q'(t)$ from $\hat{p}$ to $B(t)$. The first segment $s(t)$ of $q(t)$ will pivot around a certain vertex $\hat{p}_1$ for $t < \tau$ and around a possibly different vertex $\hat{p}_2$ for $t > \tau$. If $\hat{p}_1 = \hat{p}_2$, then the length of $q(t)$ can be decomposed into the sum of closed forms has in Lemma A.1, hence it is locally convex at τ. If $\hat{p}_1 \neq \hat{p}_2$, then we can extend the two closed forms of the length $q(t)$ before and after τ and apply Lemma A.2 to see that the length of $q(t)$ is locally convex at τ. The same can be said of the length of $q'(t)$, which implies the local convexity of d at τ. If $V(t)$ has no vertex in common before and after τ it must be that $\sigma(\tau)$ is the straight line segment between $A(\tau)$ and $B(\tau)$. The local convexity of d then follows from Corollary A.3 where $h(t)$ is the length of the line segment between $A(t)$ and $B(t)$. □

Proposition A.5 implies that the bivariate shortest-path distance function $D(u, v)$ between $A(u)$ and $B(v)$ is also convex, as follows. For any $u, u', v, v', t \in [0, 1]$ we denote by $d_{u,v,u',v'}(t)$ the shortest-path distance between $A((1 - t)u + tu')$ and $B((1 - t)v + tv')$. Said differently, we put $d_{u,v,u',v'}(t) = D((1-t)(u, v)+t(u', v'))$. Proposition A.5 implies that the univariate function $d_{u,v,u',v'}$ is convex. It follows that

$$\begin{aligned} D((1-t)(u,v) + t(u',v')) &= d_{u,v,u',v'}((1-t)\cdot 0 + t \cdot 1) \\ &\leqslant (1-t)d_{u,v,u',v'}(0) + t d_{u,v,u',v'}(1) \\ &= (1-t)D(u,v) + tD(u',v'), \end{aligned}$$

which expresses the convexity of D.

The convexity of D immediately implies the following corollary, which is the last ingredient necessary for correctness of the algorithm described in Section 5. (See Section 5 for an explanation of the notation.)

Corollary A.6. *For all integers i and j, the restriction of d_h to any grid cell $C_{i,j}$ is convex.*

Erin Wolf Chambers, Department of Computer Science, University of Illinois at Urbana-Champaign, USA

E-mail address: erinwolf@uiuc.edu

Éric Colin de Verdière, CNRS, Département d'Informatique, Ecole Normale Supérieure, Paris, France

E-mail address: Eric.Colin.de.Verdiere@ens.fr

Jeff Erickson, Department of Computer Science, University of Illinois at Urbana-Champaign, USA

E-mail address: jeffe@cs.uiuc.edu

Sylvain Lazard, INRIA Lorraine, LORIA, Nancy, France

E-mail address: lazard@loria.fr

Francis Lazarus, CNRS, Laboratoire LIS, Institut National Polytechnique de Grenoble Grenoble, France

E-mail address: Francis.Lazarus@lis.inpg.fr

Shripad Thite, Department of Computer Science, Technische Universiteit Eindhoven, The Netherlands

E-mail address: sthite@win.tue.nl