On the Discrete Fréchet Distance in a Graph

Anne Driemel
Hausdorff Center for Mathematics, Universität Bonn, Germany

Ivor van der Hoog
Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark

Eva Rotenberg
Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark

Abstract

The Fréchet distance is a well-studied similarity measure between curves that is widely used throughout computer science. Motivated by applications where curves stem from paths and walks on an underlying graph (such as a road network), we define and study the Fréchet distance for paths and walks on graphs. When provided with a distance oracle of $O(1)$ query time, the classical quadratic-time dynamic program can compute the Fréchet distance between two walks $P$ and $Q$ in a graph $G$ in $O(|P| \cdot |Q|)$ time. We show that there are situations where the graph structure helps with computing Fréchet distance: when the graph $G$ is planar, we apply existing (approximate) distance oracles to compute a $(1 + \varepsilon)$-approximation of the Fréchet distance between any shortest path $P$ and any walk $Q$ in $O(|G| \log |G|/\sqrt{\varepsilon} + |P| + \frac{|Q|}{\varepsilon})$ time. We generalise this result to near-shortest paths, i.e. $\kappa$-straight paths, as we show how to compute a $(1 + \varepsilon)$-approximation between a $\kappa$-straight path $P$ and any walk $Q$ in $O(|G| \log |G|/\sqrt{\varepsilon} + |P| + \frac{|Q|}{\varepsilon})$ time. Our algorithmic results hold for both the strong and the weak discrete Fréchet distance over the shortest path metric in $G$.

Finally, we show that additional assumptions on the input, such as our assumption on path straightness, are indeed necessary to obtain truly subquadratic running time. We provide a conditional lower bound showing that the Fréchet distance, or even its $1.01$-approximation, between arbitrary paths in a weighted planar graph cannot be computed in $O((|P| \cdot |Q|)^{1-\delta})$ time for any $\delta > 0$ unless the Orthogonal Vector Hypothesis fails. For walks, this lower bound holds even when $G$ is planar, unit-weight and has $O(1)$ vertices.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Fréchet, graphs, planar, complexity analysis

Digital Object Identifier 10.4230/LIPIcs.SoCG.2022.36


Funding Partially supported by Independent Research Fund Denmark grants 2020-2023 (9131-00044B) “Dynamic Network Analysis”.

Acknowledgements We thank David Goeckede and Petra Mutzel for useful discussions.

1 Introduction

The Fréchet distance is a popular metric for measuring the similarity between (polygonal) curves. The Fréchet distance is often intuitively defined through the following metaphor: suppose that we have two curves that are traversed by a person and their dog. Over all possible traversals by both the person and the dog, what is the minimum length of their connecting leash? The Fréchet distance has many applications; in particular in the analysis and visualization of movement data [10, 14, 31, 44]. It is a versatile distance measure that can be used for a variety of objects, such as handwriting [38], coastlines [34], outlines of geometric shapes in geographic information systems [20], trajectories of moving objects,
such as vehicles, animals or sports players [37, 39, 6, 14], air traffic [5] and also protein structures [28]. There are many variants of the Fréchet distance, some of which we also discuss further below. The two most-studied variants are the continuous and discrete Fréchet distance (based on whether the entities traverse a curve continuously or vertex-by-vertex).

Alt and Godau [2] were the first to study the Fréchet distance from a computational perspective. They studied how to compute the continuous Fréchet distance between two polygonal curves of $n$ and $m$ vertices each in $O(nm \log (n + m))$ time. Recently, this running time was improved by Buchin et al. [11] to $O(n^2 \sqrt{\log n} (\log \log n)^{3/2})$ on a real-valued pointer machine and $O(n^2 \log \log n)$ on a word RAM with word size $\Omega(\log n)$. Eiter and Manila [23] showed how to compute the discrete Fréchet distance between two polygonal curves in $O(nm)$ time, which was later improved to $O(nm (\log \log nm) / \log nm)$ by Buchin et al. [11].

Conditional lower bounds for the Fréchet distance. The above (near-) quadratic upper bound algorithms are accompanied by a series of conditional lower bounds for computing the Fréchet distance or a constant factor approximation. All these results assume the Orthogonal Vector Hypothesis (OVH) or, by extension, the strong exponential time hypothesis (SETH) [42]. Bringmann [7] shows that there is no $O(n^{2-\delta})$ algorithm, for any $\delta > 0$, for computing the (discrete or continuous) Fréchet distance between two polygonal curves of $n$ vertices each. The statement also holds for approximation algorithms with small constant approximation factor. Bringmann’s original proof uses self-intersecting curves in the plane. Later, Bringmann and Mulzer [9] showed the same conditional lower bound for intersecting curves in $\mathbb{R}^1$. Bringmann [7] also showed the following conditional lower bound tailored to the unbalanced setting where the two input curves have different complexities: given two polygonal curves of $n$ and $m$ vertices each, there is no $O((nm)^{1-\delta})$ time algorithm for computing the Fréchet distance. Recently Buchin, Ophelders and Speckmann [13] showed that (assuming OVH) there can be no $O((nm)^{1-\delta})$ time algorithm that computes anything better than a 3-approximation of the Fréchet distance for pairwise disjoint planar curves in $\mathbb{R}^2$ and intersecting curves in $\mathbb{R}^3$.

Avoiding lower bounds. These lower bounds can be circumvented whenever the input curves come from well-behaved classes of curves, such as $c$-packed curves [22, 8], $\phi$-low density curves [22], and $\kappa$-straight curves [3, 4], and in special cases when the edges of the input curves are long [26]. Another way to avoid the quadratic complexity is to allow relatively large approximation factors. Bringmann and Mulzer [9] presented an $\alpha$-approximation algorithm for the discrete Fréchet distance, that runs in time $O(n \log n + n^2/\alpha)$, for any $\alpha$ in $[1, n]$. This was recently improved by Chan and Rahmati [16] to $O(n \log n + n^2/\alpha^2)$ for any $\alpha$ in $[1, n/\log n]$. For the continuous Fréchet distance a weaker result was presented by Colombe and Fox [19]. They show an $O(\alpha)$-approximation algorithm for any $\alpha$ in $[\sqrt{n}, n]$ that runs in time $O(n^3/\alpha^2 \log n)$. For general polygonal curves, without further input assumptions, the best-known approximation factors with near-linear running times are still quite high, $\alpha \approx n$ for the continuous Fréchet distance and $\alpha \approx \sqrt{n}$ for the discrete case.

Fréchet distance variants. Variants of the Fréchet distance include those that model partial similarity by allowing straight-line shortcuts along a curve [21], or by maximizing the portions of the curves that a matched to each other within a fixed distance [12]. Other variants constrain the class of mappings by applying speed constraints [33] or topological constraints [15], or model the distance metric to the geodesics inside a simple polygon [27]. Even other variants extend the class of mappings, such as the weak Fréchet distance, which
A road network can be represented as a graph $G$. (b) Edges in $G$ can be weighted, e.g., depending on whether traffic flows fast (grey) or slow (black). Under the shortest path metric, the Fréchet distance between blue and green may be smaller than the distance between red and black; even though under the Euclidean metric, the red-black Fréchet distance is smaller.

was already studied by Alt and Godau [2]. Strikingly, the Fréchet distance has not been studied in the context of graphs. Edge-weighted graphs with their shortest-path metric are commonly used to model discrete metric spaces [35], and the Fréchet distance can be derived from the underlying distance metric (Figure 2). In this paper, we intend to initiate a study of the computational complexity of the discrete Fréchet distance between paths in a planar graph, where distances between nodes are measured by their shortest path metric in this graph. This is a natural model when, for example, measuring the similarity of two trajectories in the same street network (Figure 1).

**Contribution and organisation.** This is the first paper that considers computing the Fréchet distance in the graph domain.\(^1\) Section 2 contains the preliminaries where we present an overview of distance oracles and the problem statement. Section 3 serves as an introduction to our setting and techniques. We assume that $P$ is a $\kappa$-straight path and that $Q$ is a walk in a planar weighted graph $G$. We use an exact distance oracle with $O(\log^{2+o(1)} |G|)$ query time to compute a $(\kappa + 1)$-approximation of $D_{SF}(P, Q)$. This is the first nontrivial algorithm for computing the (approximate) Fréchet distance in a planar graph. In Section 4 we extend our results. We use a $(1 + \alpha)$-stretch distance oracle to compute a $(1 + \varepsilon)$-approximation of $D_{SF}(P, Q)$. The full version contains the analogous result for the weak Fréchet distance. Finally, we show in Section 5 a conditional lower bound for computing the Fréchet distance. Specifically, assuming the Orthogonal Vector Hypothesis (OVH), we show that if $G$ is an integer-weighted planar graph, $P$ and $Q$ are paths in $G$ and $m = n^\gamma$ for some constant $\gamma > 0$, then for every $\delta > 0$ there can be no algorithm that computes $D_{SF}(P, Q)$ (or a $1.01$-approximation) in $O((nm)^{1-\delta})$ time unless OVH fails. In the full version we consider walks $P$ and $Q$ in a planar unit-weight graph with a constant number of vertices.

**2 Preliminaries**

Let $G = (V, E)$ be a planar undirected weighted graph with $N$ vertices, where every edge $e_i$ has some corresponding integer weight $\omega_i$ and all weights can be expressed in a word of $\Theta(\log N)$ bits. For any two vertices $v_1, v_2 \in V$ their distance, denoted by $d(v_1, v_2)$, is the

---

\(^1\) Similar ideas were used in the master’s thesis of David Goeckede [24]. In particular, the approach we use in Section 3 and a lower bound construction for walks was used there.
length of the shortest path from \( v_1 \) to \( v_2 \) in \( G \). A walk in \( G \) is any sequence of vertices where every subsequent pair of vertices is connected by an edge in \( E \). A path in \( G \) is a walk where no vertex appears twice in the sequence.

Let \( P \) be any walk in \( G \), represented by an ordered set of vertices \( P = (p_1, p_2, \ldots, p_n) \). We denote by \( |P| = n \) the number of vertices in \( P \) and by \([n]\) the set \((1, 2, \ldots, n)\). We denote the walk \( Q = (q_1, q_2, \ldots, q_m) \), \(|Q|\) and \([m]\) analogously.

**Discrete Fréchet distance.** Given two walks \( P \) and \( Q \) in \( G \), we denote by \([n]\times[m] \subset \mathbb{N} \times \mathbb{N}\) the integer lattice of \( n \) by \( m \) integers. We say that an ordered sequence \( F \) of points in \([n]\times[m]\) is a discrete walk if for every consecutive pair \((i,j),(k,l) \in F\), we have \( k \in \{i-1, i, i+1\} \) and \( l \in \{j-1, j, j+1\} \). It is furthermore \( xy\)-monotone when we restrict to \( k \in \{i, i+1\} \) and \( l \in \{j, j+1\} \). Let \( F \) be a discrete walk from \((1,1)\) to \((n,m)\). The cost of \( F \) is the maximum over \((i,j) \in F\) of \( d(p_i,q_j) \). The (weak) discrete Fréchet distance is the minimum over all (not necessarily \( xy\)-monotone) walks \( F \) from \((1,1)\) to \((n,m)\) of its associated cost:

\[
D_F(P,Q) := \min_{F} \text{cost}(F) = \min_{F} \max_{(i,j) \in F} d(p_i,q_j).
\]

**The discrete free-space matrix.** In this paper we show an algorithm for computing the discrete Fréchet distance between two walks \( P \) and \( Q \) in a graph \( G \). To this end, we use what we will call a free-space matrix which can be seen as a discrete free-space diagram. Given \( P, Q \) and some real value \( \rho \), we construct a \(|P| \times |Q|\) matrix \( M_\rho \) which we call the free-space matrix of \( P \) and \( Q \). The \( i'th \) column of \( M_\rho \) corresponds to the vertex \( p_i \in P \) and the \( j'th \) row corresponds \( q_j \in Q \). We assign to each matrix cell \( M_\rho[i,j] \) the integer \(-1\) if \( d(p_i,q_j) \leq \rho \), and a 0 if \( d(p_i,q_j) > \rho \). From our above definition of the discrete Fréchet distance, we immediately conclude the following:

**Lemma 1.** The Fréchet distance between \( P \) and \( Q \) is at most \( \rho \), if and only if there exists a discrete (\( xy\)-monotone) walk \( F \) from \((1,1)\) to \((n,m)\) such that \( \forall (i,j) \in F \), \( M_\rho[i,j] = -1 \).

**Orthogonal Vectors Hypothesis.** The Orthogonal Vectors problem can be stated as follows. Given are a set \( A \) and \( B \) of \( d \)-dimensional Boolean vectors with \(|A| = n \) and \(|B| = m \). The goal is to identify whether there exist two vectors \( a = (a_1, a_2, \ldots, a_d) \) and \( b = (b_1, b_2, \ldots, b_d) \) with \( a \in A \) and \( b \in B \), such that \( a \) and \( b \) are orthogonal (i.e. \( \sum_{i=1}^{d} a_i \cdot b_i = 0 \)). In this paper, we use the following variant of the Orthogonal Vectors hypothesis. It is implied by SETH, see Abboud and Williams [1, Section 3], and it is equivalent to the standard variant of OVH defined by Williams [42], see Bringmann [7].

**Definition 2.** The Orthogonal Vectors Hypothesis states that for every \( \delta > 0 \) and \( 1 > \gamma > 0 \), there exists an \( \omega > 0 \) and such that the Orthogonal Vectors problem for \( d \)-dimensional vectors with \( d = \omega \log n \) and \( m = n^\gamma \), cannot be solved in \( O((nm)^{1-\delta}) \) time.
Distance oracles. A distance oracle is a compact data structure that facilitates fast exact or approximate distance queries between vertices in a graph. A distance oracle has stretch $S$ if it never underestimates the distance, and it at most overestimates by a factor $S$, i.e., $d(a, b) \leq d_{\text{estim}}(a, b) \leq S \cdot d(a, b)$. For general graphs [36, 41, 43], the best possible stretch in sub-quadratic space is 3, but for planar graphs on $N$ vertices, Thorup [40] shows that it is possible to compute $(1 + \varepsilon)$-stretch distance oracles in the near-linear $O(N/\varepsilon \log N)$ time and space, and with a query-time of $O(1/\varepsilon)$. The study of distance oracles for planar graphs is an active research area [17, 18, 25, 29, 30, 32, 40]. For $(1 + \varepsilon)$-stretch oracles, Gu and Xu [25] show that it is possible to achieve constant query-time independently of $\varepsilon$ at the cost of an increased construction time and space of $O\left(N(\log N)^{1/\varepsilon} + 2^{O(1/\varepsilon)}\right)$. Even for exact distances, Charalampopoulos et al. [17] give an $O\left(N^{1+\alpha(1)}\right)$-space and $O\left(N^{\alpha(1)}\right)$-query time data structure. Long and Pettie [32] improve these exact queries to polylogarithmic $O\left((\log(N))^{2+\alpha(1)}\right)$ time while maintaining the $O\left(N^{1+\alpha(1)}\right)$-space bound.

In the following sections we use the exact distance oracle by Long and Pettie [32] and the $(1 + \varepsilon)$-stretch oracle by Thorup [40]. Any distance oracle that improves the efficiency of these data structures, or any extension of them to larger classes of graphs, immediately leads to improving or extending our results correspondingly.

From distance oracles to an upper bound. Given a distance oracle with $T(G)$ query time it is straightforward to find an $O(nm \cdot T(G))$ time algorithm for computing $D_{\mathcal{F}}(P, Q)$ between two walks $P$ and $Q$ in $G$ that “matches” the conditional $\Omega(nm^{1-\delta})$ lower bound. Indeed, for any pair $(p, q) \in P \times Q$ we can query their pairwise distance in $G$. Given such a weighted graph, we want to find an $xy$-monotone path from $(1, 1)$ to $(n, m)$ with minimal cost (which can be done with an $O(nm \cdot T(G))$ dynamic program as by Eiter and Manuela [23]).

\(\kappa\)-straight paths. Alt, Knauer and Wenk [3] define $\kappa$-straight paths as a generalisation of shortest paths. A path $P$ is $\kappa$-straight if for any two points $s, t \in P$, the length of the subpath $P[s, t]$ from $s$ to $t$ is at most $\kappa \cdot d(s, t)$. Shortest paths are 1-straight. When we replace the term “points” by “vertices”, this definition immediately transfers to our graph setting.

3 A $(\kappa + 1)$-approximation for the discrete Fréchet distance

Let $G = (V, E)$ be a planar weighted graph with $N$ vertices and integer weights. We use the structure by Long and Pettie [32] to preprocess $G$, such that given two walks $P = (p_1, \ldots, p_n)$ and $Q = (q_1, \ldots, q_m)$, where $P$ is a $\kappa$-straight path we can compute a $(\kappa + 1)$-approximation of $D_{\mathcal{F}}(P, Q)$. In the following section we extend this approach to an algorithmic result for computing a $(1 + \varepsilon)$-approximation. Recall that the decision variant of the Fréchet distance may be answered with the help of a free-space matrix $M_{\rho}$. Here, we extend its definition:

\textbf{Definition 3.} We denote by $M_{\rho}^{\kappa}$ the $\kappa$-straight free-space matrix, which is a matrix with dimensions $n \times m$. We define the matrix $M_{\rho}^{\kappa}[i, j]$ as follows:

- $M_{\rho}^{\kappa}[i, j] = -1$ if the distance $d(p_i, q_j) \leq \rho$,
- $M_{\rho}^{\kappa}[i, j] = 1$ if the distance $d(p_i, q_j) > (\kappa + 1)\rho$, or
- $M_{\rho}^{\kappa}[i, j] = 0$ otherwise.

Every cell $M_{\rho}^{\kappa}[i, j]$ has a corresponding point $(i, j)$ in the integer lattice $[n] \times [m]$. The discrete Fréchet distance is at most $\rho$, if there exists a discrete walk $F$ through $[n] \times [m]$ where for every pair $(i, j) \in F$, $M_{\rho}^{\kappa}[i, j] = -1$. Explicitly constructing $M_{\rho}^{\kappa}$ takes at least $\Omega(nm)$ time. However, we show that we can use the distance oracle to implicitly traverse $M_{\rho}^{\kappa}$ to find the existence of such a discrete walk. To this end, we first show the following:
Lemma 4. Let $P$ be a $\kappa$-straight path and $Q$ a walk in $G$, $\rho$ be some fixed value and $j \leq m$ some integer. For any two integers $a, c$ such that $M^\kappa_{\rho}[a,j] = -1$ and $M^\kappa_{\rho}[c,j] = -1$, there cannot be an integer $b \in [a, c]$ for which $M^\kappa_{\rho}[a,b] = 1$.

Proof. Suppose for the sake of contradiction that there are three integers $a, b, c$ with $b \in [a, c]$, $M^\kappa_{\rho}[a,j] = -1$ and $M^\kappa_{\rho}[c,j] = -1$ and $M^\kappa_{\rho}[b,j] = 1$. It cannot be that $b = a$ or $b = c$, so there are three vertices $p_a, p_b, p_c \in P$ with $d(p_a, q_j) \leq \rho, d(p_c, q_j) \leq \rho$ and $d(p_b, q_j) > (\kappa + 1)\rho$ (Figure 3). Moreover, $p_b$ lies on the $\kappa$-straight subpath $P[p_a, p_c]$. It follows that the length of the subtrajectory $P[p_a, p_b]$ is more than $\kappa\rho$ (otherwise, the distance between $p_b$ and $q_j$ is at most $(\kappa + 1)\rho$ by the path through $p_a$ to $q_j$). We can apply a symmetric argument to $P[p_b, p_c]$. Thus, the length of $P[p_a, p_c]$ is more than $2\kappa\rho$. At the same time, there exists a path in $G$ from $p_a$ to $p_b$ through $q_j$ of length at most $2\rho$. This contradicts that $P$ is $\kappa$-straight.

A consequence of the above lemma is the following: let $(i, j)$ be a lattice point for which $M^\kappa_{\rho}[i,j] = -1$. For the nearest lattice point $(l, j)$ left of $(i, j)$ for which $M^\kappa_{\rho}[l,j] = 1$, there can be no lattice point left of $(l, j)$ for which the matrix evaluates to $-1$. A symmetrical statement holds for the nearest such point right of $(i, j)$. This leads to the following algorithm to conclude if $D_{\cal F}(P, Q) \leq (\kappa + 1)\rho$ or $D_{\cal F}(P, Q) > \rho$, where we construct a discrete walk $F'$:

We compute the distance oracle in $O(\lambda^{1+\Theta(1)})$ time. If $M^\kappa_{\rho}[1,1] > 1$ then our algorithm terminates and concludes that $D_{\cal F}(P, Q) > \rho$. We iteratively perform the following procedure, to construct a path $F'$. Let $(i, j)$ be the latest point added to $F'$, then:
1. If $(i, j) = (n, m)$ the algorithm terminates and concludes that $D_{\cal F}(P, Q) \leq (\kappa + 1)\rho$.
2. If $(j + 1) > m$, go to the last step.
3. Otherwise, we use two distance queries to check $M^\kappa_{\rho}[i, j + 1]$ and $M^\kappa_{\rho}[i + 1, j + 1]$:
   (i) If $M^\kappa_{\rho}[i, j + 1] = -1$, add $(i, j + 1)$ to $F'$.
   (ii) Else if $M^\kappa_{\rho}[i + 1, j + 1] = -1$, add $(i + 1, j + 1)$ to $F'$.
4. Otherwise, we use a distance query to check if $M^\kappa_{\rho}[i + 1, j]$:
   (i) If $(i + 1) > n$ or $M^\kappa_{\rho}[i + 1, j] = 1$, we terminate the procedure and conclude that $D_{\cal F}(P, Q) > \rho$.
   (ii) Otherwise, we add $(i + 1, j)$ to $F'$.

Figure 4 Lattice points to prove Lemma 5. Blue $\in F$. Orange $\in F'$ and Red $\notin F$. 

Figure 3 (a) Three vertices $p_a, p_b, p_c \in P$ and a vertex $q_j \in Q$ such that $M^\kappa_{\rho}[a,j] = M^\kappa_{\rho}[c,j] = -1$ and $M^\kappa_{\rho}[b,j] = 1$. (b) We show that the distance between $p_a$ and $p_b$ must be more than $\kappa\rho$. (c) However, this implies that $P$ is not $\kappa$-straight, as there is a shortcut from $p_a$ to $p_c$ through $q_j$. 

36:6 On the Discrete Fréchet Distance in a Graph
Lemma 5. Let $P$ be $\kappa$-straight in $G$, $Q$ be any walk and $D_{\mathcal{F}}(P, Q) < \rho$. Denote by $F$ an $xy$-monotone path over the lattice $[n] \times [m]$ such that for all $(i, j) \in F$, $M[i, j] = -1$. All lattice points in our constructed path $F'$ are either in $F$ or lie to the left of a point of $F$.

Proof. Consider for the sake of contradiction the first iteration where the algorithm would add a lattice point $(c, d)$ right of a point in $F$. Let $(a, b) \in F'$ be the point preceding $(c, d)$. We make a case distinction based on whether $(c, d)$ was added through step 3(i), 3(ii) or 4(ii).

The three cases are illustrated by Figure 4, (a) (b) and (c) respectively.

First suppose that $(c, d) = (a, b + 1)$. Since $(c, d)$ is the first point right of $F$, it must be that $F$ contains either $(a, b)$ or a point right of $(a, b)$. Moreover (since $(c, d)$ is right of $F$), $F$ also contains a point left of $(a, b + 1)$. This implies that $F$ is not $xy$-monotone, contradiction.

Now suppose that $(c, d) = (a + 1, b + 1)$. Because we reached step 3(ii), we know that $M^\rho_{\mathcal{F}}[a, b + 1] > -1$ and thus $(a, b + 1) \notin F$. However, since $(c, d)$ is the first point right of $F$, $F$ either contains $(a, b)$ or a point right of $(a, b)$, and a point strictly left of $(a, b + 1)$. This implies that $F$ is not $xy$-monotone which is a contradiction.

Finally, suppose that $(c, d) = (a + 1, b)$. Since $(c, d)$ is the first point right of $F$, it must be that $(a, b) \notin F$. However, consider now the successor of $(a, b)$ in $F$. Since $F$ is $xy$-monotone, this successor is either $(a, b + 1)$ or $(a + 1, b + 1)$, as it cannot be $(a + 1, b) = (c, d)$. However, this implies that either $M^\rho_{\mathcal{F}}[a, b + 1] = -1$ or $M^\rho_{\mathcal{F}}[a + 1, b + 1] = -1$, which contradicts the assumption that we have reached step 4 of the algorithm. \hfill \qed

With these two observations, we are ready to prove our main theorem:

Theorem 6. We can preprocess a planar graph $G$ with $N$ vertices in $O(N^{1+o(1)})$ time and space s.t: for any $\kappa$-straight path $P = (p_1, \ldots, p_n)$, walk $Q = (q_1, \ldots, q_m)$ and $\rho \in \mathbb{R}$, we can conclude either $D_{\mathcal{F}}(P, Q) > \rho$ or $D_{\mathcal{F}}(P, Q) \leq (\kappa + 1)\rho$ in $O((n + m)\log^{2+o(1)} N)$ time.

Proof. We first preprocess $G$ to construct a distance oracle using $O(N^{1+o(1)})$ time and space. Given $\rho$, our algorithm spends at most $n + m$ iterations before it either reaches $(n, m)$ or step 4(i) and terminates. At each iteration we perform at most three distance queries. We prove that if $D_{\mathcal{F}}(P, Q) \leq \rho$, we always conclude that $D_{\mathcal{F}}(P, Q) \leq (\kappa + 1)\rho$. Indeed, suppose that $D_{\mathcal{F}}(P, Q) \leq \rho$ then there exists a discrete walk $F$ such that for every $(i, j) \in F$, $M^\rho_{\mathcal{F}}[i, j] = -1$ and $F$ is $xy$-monotone. Per construction, the path $F'$ is $xy$-monotone and for all $(i, j) \in F'$, $M[i, j] < 1$. What remains to show is that $F'$ is from $(1, 1)$ to $(n, m)$. Suppose for the sake of contradiction that $F'$ does not reach $(n, m)$ and let $(i, j)$ be the last element added to $F'$ before the algorithm terminated in step 4. Since we reached step 4 it must be that:

$M^\rho_{\mathcal{F}}[i, j + 1] > -1$ and $M^\rho_{\mathcal{F}}[i + 1, j + 1] > -1$ (or $(j + 1 \leq m)$).

Let $\ell \leq i$ be the lowest integer such that $M^\rho_{\mathcal{F}}[\ell, j] = -1$. Such an $\ell$ must always exist, since we only enter the $j$'th row through a point $(k, j)$ for which $M^\rho_{\mathcal{F}}[k, j] = -1$ (step 3(ii) or 3(iii)).

Since we arrived in step 4(ii), it must be that either $M^\rho_{\mathcal{F}}[i + 1, j] = 1$ or $(i + 1) > n$. However, this implies that $(i, j) \in F$ (indeed, by Lemma 5 there exists a point equal to or to the right of $(i, j)$ in $F$). However, given Lemma 4 and $(\ell, i)$, there is no a point in $F$ right of $(i, j)$. Because if $F$ is $xy$-monotone, the successor of $(i, j) \in F$ is either $(i + j, 1)$, $(i, j + 1)$ or $(i, j + 1)$. Since we terminated, none of these elements can be in $F$, contradiction. \hfill \qed

The following corollary is a direct result of the assumption that edge weights each fit in a constant number of words (thus, the range of values for $D_{\mathcal{F}}(P, Q)$ is terminated in constant time).

Corollary 7. We can preprocess a planar graph $G$ with $N$ vertices in $O(N^{1+o(1)})$ time such that: for any $\kappa$-straight path $P = (p_1, \ldots, p_n)$ and walk $Q = (q_1, \ldots, q_m)$, we can compute a $(\kappa + 1)$-approximation of $D(G)(P, Q)$ in $O((n + m)\log^{3+o(1)} N)$ time.
4 A \((1 + \varepsilon)\)-approximation for Fréchet distance

We present a more involved approach to compute a \((1 + \varepsilon)\) approximation of \(D_F(P, Q)\). Specifically, we choose \((1 + \varepsilon) = (1 + \alpha)(1 + \alpha + \beta)\) for some \(\alpha\) and \(\beta\). We show for any \(\rho\) how to correctly conclude either \(D_F(P, Q) \leq (1 + \alpha)(1 + \alpha + \beta)\rho\) or \(D_F(P, Q) > \rho\).

To obtain this result, we use two data structures. A Voronoi diagram of \(P\) in \(G\) marks every vertex \(v\) in \(G\) with the closest vertex \(p \in P\) (and the exact distance \(d(v, p)\)). For completeness, we prove in the full version the following (folklore) result:

\[\text{Lemma } 10.\] For any planar weighted graph \(G = (V, E)\) and any vertex set \(P \subseteq V\), it is possible to construct the Voronoi diagram of \(P\) in \(G\) in \(O(|V| \log |V|)\) time.

Additionally, we use the \((1 + \alpha)\)-stretch distance oracle \(\mathcal{D}(G)\) by Thorup [40]. We differentiate between the distance \(d(p_i, q_j)\) and what we call the perceived distance between \(p_i\) and \(q_j\). For any two vertices \(p_i, q_j\) we denote by \(d_o(p_i, q_j)\) their perceived distance (the result of the distance query of \(\mathcal{D}(G)\)). Per definition \(d(p_i, q_j) \leq d_o(p_i, q_j) \leq (1 + \alpha) \cdot d(p_i, q_j)\).

\[\text{Definition } 9.\] For a given \(\rho \in \mathbb{R}\) we denote by \(M^\rho\) the approximate free-space matrix, which is a matrix with dimensions \(n \times m\) where:

- \(M^\rho[i, j] = -1\) if the perceived distance \(d_o(p_i, q_j) \leq (1 + \alpha)\rho\),
- \(M^\rho[i, j] = 1\) if the perceived distance \(d_o(p_i, q_j) > (1 + \alpha)(1 + \alpha + \beta)\rho\), or
- \(M^\rho[i, j] = 0\) otherwise.

\(\beta\)-compression. Given a \(\kappa\)-straight path \(P\) and real values \((\rho, \beta)\) we define the \(\beta\)-compression \(P^\beta\) as an ordered set that is obtained in three steps (Figure 5):

- The first step is a greedy iterative process where:
  - we remove (consecutive) \(p_x\) where the length of \(P[p_{x-1}, p_x]\) is fewer than \(\beta\rho\).
  - the first such vertex \(p_{x-1}\) that does not meet this criterion is added to \(P^\beta\). Then, we remove (consecutive) \(p_x\) where the length of \(P[p_{x-1}, p_x]\) is fewer than \(\beta\rho\) and so forth.
  - In the second step we add for every vertex in \(P^\beta\) its preceding vertex in \(P\).
  - In the third step we add \(p_n\).

The result of this procedure is that we have an ordered set \(P^\beta\) with \(n' \leq n\) vertices. We create a map \(\pi : [n'] \rightarrow [n]\) that maps every vertex in \(P^\beta\) to its corresponding vertex in \(P\) (i.e. the \(k\)th element of \(P^\beta\) is denoted by \(p_{\pi(k)} \in P\)) and we observe:

- \(\pi(1) = 1\) and \(\pi(n') = n\),
- for all \(i\), the length of \(P[p_{\pi(i)}, p_{\pi(i+\beta)}]\) is greater than \(\beta\rho\) and
- for all \(x \in [\pi(i), \pi(i+1)]\), the exact distance \(d(p_{\pi(i)}, p_x) < \beta\rho\) and \(d(p_{\pi(i+1)}, p_x) < \beta\rho\).

We denote \(P^\beta = (p_{\pi(1)}, p_{\pi(2)}, \ldots p_{\pi(n')})\). The global approach is to approximate the Fréchet distance between \(P^\beta\) and \(Q\) instead. We first note the following three properties of \(P^\beta\):

\[\text{Lemma } 10.\] For every two integers \(i\) and \(j\), if \(M^\rho[\pi(i), j] = -1\), then for all integers \(x \in (\pi(i-1), \pi(i+1))\) it must be that \(M^\rho[i, j] \leq 1\).

\[\text{Proof.}\] Either \(p_{\pi(i-1)}\) and \(p_{\pi(i)}\) are consecutive in \(P\) (thus, the set \((\pi(i-1), \pi(i))\) is empty) or per construction the length of \(P[p_{\pi(i-1)}, p_{\pi(i)}]\) is less than \(\beta\rho\).

Thus, if the perceived distance \(d_o(p_{\pi(i-1)}, q_j) \leq (1 + \alpha)\rho\), then for all points \(p_x\) with \(x \in (\pi(i-1), \pi(i))\), the exact distance \(d(p_x, q_j) \leq (1 + \alpha + \beta)\rho\) by traversing through \(p_{\pi(i)}\). Thus, the perceived distance \(d_o(p_x, q_j) \leq (1 + \alpha)(1 + \alpha + \beta)\rho\). A symmetrical argument holds for all \(x \in (\pi(i), \pi(i+1))\). \(\Box\)
Figure 5 A planar path where the edge weights correspond to their length. (a) We greedily add vertices to $P^3$ such that for all vertices $p_x \in P$ with preceding vertex $p_i \in P^3$, the length of $P[p_i, p_x]$ is at most $\beta \rho$. (b) For every vertex in $P^3$, we subsequently add its preceding vertex in $P$ to $P^3$.

Lemma 11. For all $i$ and $j$, if there exists an integer $x \in (\pi(i), \pi(i+1))$ such that $M_\beta^3[x, j] = -1$, then $M_\beta^3[\pi(i), j] \leq 1$ and $M_\beta^3[\pi(i+1), j] \leq 1$.

Proof. As in Lemma 10, $d(p_x, p_{\pi(i)}) \leq \beta \rho$ and $d(p_x, p_{\pi(i+1)}) \leq \beta \rho$ implies the lemma.

Lemma 12. For any $j$, let $i$ be an integer such that there exists an $x \in [\pi(i), \pi(i+1)]$ with $M_\beta^3[x, j] = -1$. Denote $a = i - \lceil \frac{3\kappa}{\beta} \rceil$ and $b = i + \lceil \frac{3\kappa}{\beta} \rceil$. There can be no integer $y \notin [\pi(a), \pi(b)]$ such that $M_\beta^3[y, j] = -1$.

Proof. For all $i$, the length of $P[p_{\pi(i)}, p_{\pi(i+3)}]$ is greater than $\beta \rho$. It follows that the length of the subpath $P[p_{\pi(a)}, p_x]$ is more than $\sum_{x=1}^{\lceil 3\kappa/\beta \rceil} \beta \rho = \frac{3\kappa}{\beta} \beta \rho = 3\kappa \rho$ (Figure 6). Suppose for the sake of contradiction that there exists an integer $y < \pi(a)$ such that $d_\omega(p_y, q_j) \leq (1+\alpha)\rho$. Then the exact distance $d(p_y, p_x)$ is at most $2(1+\alpha)\rho$ through traversing from $p_y$ to $q_j$ to $p_x$.

However, the subpath $P[p_y, p_x]$ is longer than $P[p_{\pi(a)}, p_x]$ and thus longer than $3\kappa \rho$. For $\alpha < 0.5$, this contradicts the assumption that $P$ is $\kappa$-straight. A symmetrical argument holds for $y > \pi(b)$.

Figure 6 A schematic representation of $P^3$. For any $i$ as in Lemma 12, we consider an integer $a = i - \lceil \frac{3\kappa}{\beta} \rceil$ and some $p_y$ preceding $p_{\pi(a)}$. 

\[ \pi(7) = 12 \]
Defining $\beta$-windows. Now, we use two lattices: $[n] \times [m]$ and the smaller lattice $[n'] \times [m]$. Points on the first lattice will be denoted by $(x,j)$ and $(y,j)$. Points on the second lattice will be denoted by $(i,j)$ or $(a,j)$ or $(b,j)$. Intuitively, Lemma 12 shows for every integer $j$ a “horizontal window” in $[n'] \times [m]$ (of width $O(\frac{\alpha}{\beta})$) that bounds the subpath of $P$ of vertices that may have perceived distance fewer than $(1+\alpha)\rho$ to the vertex $q_j \in Q$. We formalise this intuition by defining $\beta$-windows (denoted by $W_1, W_2, \ldots, W_m$, see Figure 7):

- Let for an index $j$, $p_x$ be any vertex in $P$ with minimal distance to $q_j$ in the graph $G$.
- Let $i$ be the integer such that $p_{\pi(i)}$ is the point in $P^\beta$ that precedes $p_x$.
- We distinguish two cases:
  1. If the exact distance $d(p_x, q_j) > \rho$ then: $W_j$ is empty.
  2. Otherwise: $W_j = [i - \lfloor \frac{\alpha \rho}{\beta} \rfloor, i + \lfloor \frac{\alpha \rho}{\beta} \rfloor] \times \{j\} \subset [n'] \times [m]$.

The high-level approach. We first construct the Voronoi diagram of $P$ in $G$ in $O(N \log N)$ time. For every $q_j \in Q$, we obtain from the diagram the vertex $p_x$ with minimal distance to $q_j$ in the graph $G$. For every point $(a,j) \in W_j$, we compute $d(p_{\pi(a)}, j)$ in $O(\frac{1}{\alpha})$ time. Any lattice walk that realises a distance $D_F(P, Q) \leq (1+\alpha)(1+\alpha + \beta)\rho$ must be contained in the grid: $A = \cup_j W_j$ which has $O(m \cdot \frac{\alpha}{\beta})$ complexity. We compute a minimal cost path in time linear in the size of $A$.

$P^\beta$

(a)

(b)

(c)

Figure 7 (a) a schematic representation of a path $P$ with $P^\beta$ in red. (b) For every $j \in [m]$, we observe the closest point $p_x$. If $d(p_x, q_j) \leq \rho$ we color it green. Otherwise, we color it orange. In addition, if $p_x \notin P^\beta$ we color its predecessor in $P^\beta$ yellow. (c) For every yellow or green vertex in $[n'] \times [m]$, we create a horizontal window in blue. We show the window for $\kappa = \beta = 1$.

Theorem 13. Let $G$ be a planar graph with $N$ vertices, $P = (p_1, \ldots, p_n)$ a $\kappa$-straight path and $Q = (q_1, \ldots, q_m)$ be any walk in $G$. Given a value $\rho \in \mathbb{R}$ and some $\beta$ and $\alpha \leq 0.5$, we correctly conclude either $D_F(P, Q) > \rho$ or $D_F(P, Q) \leq (1+\alpha)(1+\alpha + \beta)\rho$ in $O(N \log N/\alpha + n + \frac{\alpha}{\alpha^2} m)$ time using $O(N \log N/\alpha)$ space.
Proof. We construct the approximate distance oracle $\mathcal{D}(G)$ using $O(N \log N/\alpha)$ time and space. Given $P$ and $Q$, we construct the $\beta$-compressed path $P^\beta$ in $O(n)$ time. We supply every point in $P \setminus P^\beta$ with a pointer to the point in $P^\beta$ that precedes it. We construct the Voronoi diagram of $P$ in the graph $G$ in $O(N \log N)$ time. Given $P^\beta$, we construct for every integer $j \in [m]$ the window $W_j$ in $O(\frac{N}{\beta})$ time. Specifically, for any point $q_j$ we obtain the point $p_x$ that is closest to $q_j$. If $d(p_x, q_j) \leq \rho$ then we obtain the point $q_{\pi(i)}$ in $P^\beta$ that precedes $p_x$ in constant time through the pre-stored pointer and we set: $W_j = [i - \left\lfloor \frac{2n}{\beta} \right\rfloor, i + \left\lceil \frac{2n}{\beta} \right\rceil] \times \{j\}$.

The union of windows $(A = \cup W_j)$ is a grid in $[n'] \times [m]$ of at most $O(m \cdot \frac{N}{\beta})$ lattice points. For each $(a, j) \in A$ we query $\mathcal{D}(G)$ in $O(\frac{N}{\beta})$ time to determine the value $M^\beta_{\rho}[\pi(a), j]$ in $O(m \cdot \frac{N}{\beta})$ total time. Given this grid, we construct a directed grid graph where there is:

- a vertical edge from $(a, j)$ to $(a, j + 1)$ if $M^\beta_{\rho}[\pi(a), j] < 1$ and $M^\beta_{\rho}[\pi(a), j + 1] < 1$,
- a horizontal edge from $(a, j)$ to $(a + 1, j)$ if $M^\beta_{\rho}[\pi(a), j] < 1$ and $M^\beta_{\rho}[\pi(a + 1), j] = -1$,
- a diagonal edge from $(a, j)$ to $(a + 1, j + 1)$ if $M^\beta_{\rho}[\pi(a), j] < 1$ and $M^\beta_{\rho}[\pi(a + 1), j + 1] = -1$.

We can determine if there exists a path in $A$ from $(1, 1)$ to $(n', m)$ in $O(\frac{N}{\beta})$ time.

If such a path $F^*$ exists. we claim that $D_\mathcal{D}(P, Q) \leq (1 + \alpha)(1 + \alpha + \beta)\rho$. Indeed, we transform $F^*$ into a path over $[n] \times [m]$ as follows: for all $(a, j) \in F^*$ we add $(\pi(a), j)$. Note that per construction of the grid graph, for all points in $F^*$ it must be that $M^\beta_{\rho}[\pi(a), j] < 1$ and thus $d_\rho(\pi(a), j) \leq (1 + \alpha)(1 + \alpha + \beta)\rho$. For every two consecutive points $(a, j), (a + 1, j')$ in $F^*$, per construction, $M^\beta_{\rho}[\pi(a + 1), j'] < 1$. We add all points $(x, j')$ with $x \in [\pi(a), \pi(b)]$. By Lemma 10, for all these points $(x, j')$ it must be that $M^\beta_{\rho}[x, j'] < 1$. Thus, we found a walk $F$ from $(1, 1)$ to $(n, m)$ where for every $(i, j) \in F$, $M^\beta_{\rho}[i, j] < 1$ and the Fréchet distance between $P$ and $Q$ is at most $(1 + \alpha)(1 + \alpha + \beta)\rho$.

If no such path $F^*$ exists. we claim that $D_\mathcal{D}(P, Q) > \rho$. Suppose for the sake of contradiction that $D_\mathcal{D}(P, Q) \leq \rho$ then there exists an $xy$-monotone path $F$ from $(1, 1)$ to $(n, m)$ where for all $(i, j) \in F$, $d(p_i, q_j) \leq \rho$. We use $F$ to construct a path $F^*$ from $(1, 1)$ to $(n', m)$ in our grid graph. Specifically, for every element $(x, j) \in F$ we check if $p_x$ has been removed during compression.

- If $p_x$ has an equivalent in $P^\beta$ then there exists an integer $a$ such that $\pi(a) = p_x$ and we add the lattice point $(a, j) \in [n'] \times [m]$ to $F^*$. Per definition of $F$, $M^\beta_{\rho}[\pi(a), j] = -1$.
- Otherwise, we identify the index $i$ such that $\pi(i)$ is the vertex of $P^\beta$ preceding $p_x$ and we add the point $(i, j) \in [n'] \times [m]$ to $F^*$. By Lemma 11, $M^\beta_{\rho}[\pi(i), j] < 1$.

Since $F$ is a connected $xy$-monotone path from $(1, 1)$ to $(n, m)$, we obtain an $xy$-monotone path $F^*$ from $(1, 1)$ to $(n', m)$. Moreover, whenever this path traverses a horizontal or diagonal edge to a point $(a, j)$ it must be that $(\pi(a), j) \in F$ and thus $M^\beta_{\rho}[\pi(a), j] = -1$. Thus, $F^*$ is a path from $(1, 1)$ to $(n', m)$ in our grid graph which contradicts the earlier assumption that no such path exists.

This corollary follows immediately from choosing $\alpha = \beta = 0.25(\sqrt{8\varepsilon + 9} - 3)$.

▶ Corollary 14. Let $G$ be a planar graph with $N$ vertices, $P = (p_1, \ldots, p_n)$ a $\kappa$-straight path and $Q = (q_1, \ldots, q_m)$ be any walk in $G$. Given a value $\rho \in \mathbb{R}$ and some $\varepsilon > 0$ we correctly conclude either $D_\mathcal{D}(P, Q) > \rho$ or $D_\mathcal{D}(P, Q) \leq (1 + \varepsilon)\rho$ in $O(N \log N/\sqrt{\varepsilon} + n + \frac{\varepsilon}{\kappa}m)$ time.

5 A conditional lower bound for computing the Fréchet distance

We show that for every $\delta > 0$ there is no $O((nm)^{1-\delta})$ algorithm for computing the discrete Fréchet distance between two paths in a planar graph (unless O VH fails). We show this using a planar graph $G = (V, E)$ where the edges have integer weights in $\{0.001, 0.35, 0.6, 0.65, 1, 2, 3\}$.
In the full version we prove a similar statement for walks in a constant-complexity unit-weight graph. Throughout this section, we fix some $\delta > 0$ and $\gamma > 0$ and consider two sets $A$ and $B$ of $d$-dimensional Boolean vectors (with $d = \omega \log n$ where the constant $\omega$ depends on $\delta$). In addition, we assume that $A$ and $B$ contain $n'$ and $m'$ vectors respectively with $n' = (m')^\gamma$.

Using $A$ and $B$, we reduce from Orthogonal Vectors using what we call a vector gadget. We construct a graph $G$ and two paths $P$ and $Q$ where $D_F(P, Q) < 3$ if and only if there exists $(a, b) \in A \times B$ such that $a$ and $b$ are orthogonal.

**Proof notation.** Throughout this section, we label vertices to represent an equivalence class.

We construct a graph where we label "blue" vertices with a label in $\{x, y, z, B^{(0)}, B^{(1)}\}$ and "red" vertices with a label in $\{\alpha, \alpha^*, \beta, \beta^*, \gamma, A^{(0)}, A^{(1)}, A\}$. Ideally, we would construct a graph where for every red-blue pair of labels, all red-blue vertices with those two labels have the same distance. We maintain a slightly weaker property: consider any red-blue pair of vertices $b, r$ with $\text{label}(b) \in \{x, y, z, B^{(0)}, B^{(1)}, B\}$ and $\text{label}(r) \in \{\alpha, \alpha^*, \beta, \beta^*, \gamma, A^{(0)}, A^{(1)}, A\}$. We demand the following: if $d(b, r) < 3$ then for all $(b', r')$ with $\text{label}(b') = \text{label}(b)$ and $\text{label}(r') = \text{label}(r)$ it must be that $d(b', r') < 3$.

We construct for every vector in $A$ (and $B$) a vector gadget. This gadget resembles the gadget used in the conditional lower bound for the Fréchet distance in the Euclidean plane by Bringmann [7]. The path $P$ will traverse all vector gadgets of $A$ in sequence (and $Q$ will traverse gadgets of $B$). We connect all gadgets of $A$ to all gadgets of $B$ via "star" vertices (grey triangles or diamonds). These stars ensure that there can be a matching between every pair of gadgets (vectors). Finally, we add "park" vertices (square vertices) which are vertices of $A$ (or $B$) that are close to all vertices of $B$ (or $A$). The intuition is, that during a traversal (reparametrization) of $P$ and $Q$ an entity can remain stationary at a park vertex, whilst the other entity traverses their corresponding path until the appropriate gadgets can be matched.

**Vector gadget.** We illustrate the vector gadget for vectors $b \in B$ (see Figure 8). The "core" of this subgraph is vertex $y$ connected to the following construction (repeated $d$ times): there are two Boolean vertices ($B^{(0)}, B^{(1)}$), followed by an intermediary vertex $B$. This core will allow us to model a $d$-dimensional Boolean vector. We connect the core to two park vertices $x$ and $z$ where we add an edge $(x, y)$ and $(B, z)$ of weight 3. Finally, we add two star vertices where every vertex $B, y$ and $x$ get connected to the top star vertex, and every vertex $x, B^{(0)}, z$ get connected to the bottom star vertex. For every vector in $A$, the corresponding vector gadget is nearly identical. Most crucially, this subgraph is vertically mirrored and the edges attached to star vertices have different weights.

**From gadgets to a graph.** Given our instance of OV, we construct $(n + m)$ vector gadgets. Next, we combine the gadgets (Figure 9). We highlight the important steps: all the vector gadgets of $B$ (and $A$) are placed horizontally adjacent to each other.

The vertices $\{s^+, z, \sigma^+\}$ get connected via a star vertex in the centre of the graph. Each vertex $s^+$ gets connected to a star vertex at the top of the graph. Each vertex $\sigma^+$ gets connected to a star vertex at the bottom of the graph. These two stars get connected via an edge with weight 2. Given this graph $G$, we say that a red vertex $r$ is close to a blue vertex $b$ if $d(r, b) < 3$. For every blue label, we observe the set of close red labels (Table 1):

**Constructing the paths $P$ and $Q$.** Given $G$, $A$ and $B$, we construct a path $P$ consisting of $n = O(n' \cdot d)$ vertices and a path $Q$ consisting of $m = O(m' \cdot d)$ vertices (refer to Figure 9). The path $P$ starts in $\alpha$ and then moves to $\alpha^*$. Then, $P$ traverses every vector gadget of $A$ in
sequence. Let $v$ be the first vector in $A$. The path $P$ arrives at $y$ and traverses the Boolean vertices and intermediate vertices in an alternating manner (where $P$ traverses $A^{(0)}$ if the corresponding Boolean in $v$ is false and $A^{(1)}$ if the corresponding Boolean is true). Having traversed every vector gadget, $P$ moves through $\beta^*$ to $\beta$. The path $Q$ traverses every vector gadget of $B$ in sequence. Let a gadget correspond to a vector $v' \in B$:

The path $Q$ starts at the vector $x$ in the gadget and then traverses the Boolean vertices and intermediate vertices in an alternating manner (where $Q$ traverses $B^{(0)}$ if the corresponding Boolean in $v'$ is false and $B^{(1)}$ if the corresponding Boolean is true). The path $Q$ ends at the vector $z$, and continues to the next gadget.

**Theorem 15.** Let $G$ be a planar, integer-weighted graph, $P$ and $Q$ be two paths in $G$ with $n$ and $m$ vertices and $n = m^3$ for some constant $0 < \gamma \leq 1$. For all $\delta > 0$, there can be no algorithm that computes (a $1.01$-approximation of $D_F(P, Q)$) in $O((nm)^{1-\delta})$ time.

**Proof.** For any given $A$ and $B$ of $n'$ and $m'$ vectors, we construct two paths $P$ and $G$ with $n = O(n' \log n')$ and $m = O(m' \log m')$ vertices respectively. OVH postulates that there exists no algorithm that can conclude if there exists two orthogonal vectors $(a, b) \in A \times B$ in $O((nm)^{1-\delta})$ time, for any $\delta > 0$. We prove this theorem by showing that there are two such vectors if and only if $D_F(P, Q) < 3$. We observe that in our graph for all red/blue vertices $r$ and $b$ either $d(r, b) \leq 2.96$ or $d(r, b) \geq 3$ (which implies this proof for the $1.01$-approximation).
Figure 9

Top: we show how pairwise gadgets get connected. Bottom: given a set $A$ of four and $B$ of three vectors, we construct the corresponding graph and path.
We show that if there exist two orthogonal vectors \((a, b) \in A \times B\) then \(D_F(P, Q) < 3\). We construct a traversal of \(P\) and \(Q\) where the red entity (henceforth “Red”) traversing \(P\) remains close to the blue entity (“Blue”) traversing \(Q\). First, Red is stationary at the park vertex \(\alpha\), whilst Blue traverses \(B\) until it reaches the vector gadget corresponding to \(b \in B\). Then, whilst Blue remains stationary at the park vertex \(x\), Red traverses \(P\) until it reaches the vector gadget corresponding to \(a \in A\). At this point, Blue moves to \(y\) as Red moves to \(\gamma\). Both entities simultaneously traverse their vector gadgets. During this traversal (since \(a\) and \(b\) are orthogonal) the entities remain close. Then, Blue remains stationary at \(z\), whilst Red traverses the rest of \(P\). Finally, Red remains at \(\beta\) whilst Blue traverses the rest of \(Q\).

We show that if \(D_F(P, Q) < 3\) then there exists a pair of vectors \((a, b) \in A \times B\) such that \(a\) and \(b\) are orthogonal. Indeed, fix any traversal of \(P\) and \(Q\) that realises the Fréchet distance. When Red is at \(\alpha^*\), Blue must be at some vertex \(x\).

Consider now the time when Blue moves from \(x\) to \(y\) (where \(y\) lies in a gadget corresponding to some vector \(b \in B\)). At this time, Red cannot be at the park vertex \(\alpha\) because \(\alpha\) precedes \(\alpha^*\). Similarly, Red cannot be at the park vertex \(\beta\) because \(\beta^*\) precedes \(\beta\) (and \(\beta^*\) is not close to \(x\)). Since \(\text{close}(y) = \{\gamma, \alpha, \beta\}\), it must be that Blue is at some vertex \(\gamma\) (corresponding to some vector \(a \in A\)). Now consider the next time step, when we assume that Red moves to \(\{A^{(0)}, A^{(1)}\}\) (the argument for when Blue moves to \(\{B^{(0)}, B^{(1)}\}\) is symmetrical). If Red moves to \(A^{(0)}\) then, via the same argument as above, Blue has to simultaneously move to \(B^{(0)}\) or \(B^{(1)}\). If Red moves to \(A^{(1)}\) then Blue must move to \(B^{(0)}\). For the next time step, via the same argument, both entities must move to \(A\) and \(B\). We can continue this same argument, which shows that the two vectors \(a\) and \(b\) must be orthogonal.

6 Concluding remarks

This paper is the first to study the natural question of computing the Fréchet distance between walks \(P\) and \(Q\) in graphs. Our algorithmic results (including the Voronoi diagram construction) do not depend on the planarity of \(G\); we rely only on a distance oracle. Hence, our result immediately holds for other classes of graphs where it is possible to efficiently construct distance oracles or in computational models where the distance oracle is provided. Given a distance oracle, our \((\kappa + 1)\) approximation is obtained in time (near-) linear in \(\log(n)\). In other words, our result in Section 3 allows us to pre-process a graph \(G\) in time nearly linear to its vertices, in order to efficiently facilitate Fréchet distance queries between two any two walks in (as long as one of the two walks is \(\kappa\) straight for some query constant \(\kappa\)). This is not true for our \((1 + \epsilon)\)-approximation algorithm, which currently requires the construction of a Voronoi diagram of \(P\) in \(G\) and thus, for every pair of walks, must spend near-linear time in \(G\).

References

<table>
<thead>
<tr>
<th>No.</th>
<th>Reference</th>
</tr>
</thead>
</table>


