Why walking the dog takes time: Fréchet distance has no strongly subquadratic algorithms unless SETH fails

Karl Bringmann*

August 1, 2014

Abstract

The Fréchet distance is a well-studied and very popular measure of similarity of two curves. Many variants and extensions have been studied since Alt and Godau introduced this measure to computational geometry in 1991. Their original algorithm to compute the Fréchet distance of two polygonal curves with \( n \) vertices has a runtime of \( O(n^2 \log n) \). More than 20 years later, the state of the art algorithms for most variants still take time more than \( O(n^2 / \log n) \), but no matching lower bounds are known, not even under reasonable complexity theoretic assumptions.

To obtain a conditional lower bound, in this paper we assume the Strong Exponential Time Hypothesis or, more precisely, that there is no \( O^*((2 - \delta)^N) \) algorithm for CNF-SAT for any \( \delta > 0 \). Under this assumption we show that the Fréchet distance cannot be computed in strongly subquadratic time, i.e., in time \( O(n^{2-\delta}) \) for any \( \delta > 0 \). This means that finding faster algorithms for the Fréchet distance is as hard as finding faster CNF-SAT algorithms, and the existence of a strongly subquadratic algorithm can be considered unlikely.

Our result holds for both the continuous and the discrete Fréchet distance. We extend the main result in various directions. Based on the same assumption we (1) show non-existence of a strongly subquadratic 1.001-approximation, (2) present tight lower bounds in case the numbers of vertices of the two curves are imbalanced, and (3) examine realistic input assumptions (c-packed curves).

*Max Planck Institute for Informatics, Campus E1 4, 66123 Saarbrücken, Germany; karl.bringmann@mpi-inf.mpg.de. Karl Bringmann is a recipient of the Google Europe Fellowship in Randomized Algorithms, and this research is supported in part by this Google Fellowship.
1 Introduction

Intuitively, the (continuous) Fréchet distance of two curves $P, Q$ is the minimal length of a leash required to connect a dog to its owner, as they walk along $P$ or $Q$, respectively, without backtracking. The Fréchet distance is a very popular measure of similarity of two given curves. In contrast to distance notions such as the Hausdorff distance, it takes into account the order of the points along the curve, and thus better captures the similarity as perceived by human observers [3].

Alt and Godau introduced the Fréchet distance to computational geometry in 1991 [5, 24]. For polygonal curves $P$ and $Q$ with $n$ and $m$ vertices\(^{1}\), respectively, they presented an $O(nm \log(nm))$ algorithm. Since Alt and Godau’s seminal paper, Fréchet distance has become a rich field of research, with various directions such as generalizations to surfaces (see, e.g., [4]), approximation algorithms for realistic input curves ([6, 7, 21]), the geodesic and homotopic Fréchet distance (see, e.g., [15, 17]), and many more variants (see, e.g., [11, 20, 29, 31]). Being a natural measure for curve similarity, the Fréchet distance has found applications in various areas such as signature verification (see, e.g., [32]), map-matching tracking data (see, e.g., [9]), and moving objects analysis (see, e.g., [12]).

A particular variant that we will also discuss in this paper is the discrete Fréchet distance. Here, intuitively the dog and its owner are replaced by two frogs, and in each time step each frog can jump to the next vertex along its curve or stay at its current vertex. Defined in [22], the original algorithm for the discrete Fréchet distance has runtime $O(nm)$.

Recently, improved algorithms have been found for some variants. Agarwal et al. [2] showed how to compute the discrete Fréchet distance in (mildly) subquadratic time $O(n^2 \log n \log \log n / \log n)$. Buchin et al. [13] gave algorithms for the continuous Fréchet distance that run in time $O(n^2 \sqrt{\log n} (\log \log n)^{3/2})$ on the Real RAM and $O(n^2 (\log \log n)^2)$ on the Word RAM. However, the problem remains open whether there is a strongly subquadratic\(^{2}\) algorithm for the Fréchet distance, i.e., an algorithm with runtime $O(n^2 - \delta)$ for any $\delta > 0$. For a particular variant, the discrete Fréchet distance with shortcuts, strongly subquadratic algorithms have been found recently [8], however, this seems to have no implications for the classical continuous or discrete Fréchet distance.

The only known lower bound shows that the Fréchet distance takes time $\Omega(n \log n)$ (in the algebraic decision tree model) [10]. The typical way of proving (conditional) quadratic lower bounds for geometric problems is via 3SUM [23], in fact, Alt conjectured that the Fréchet distance is 3SUM-hard. Buchin et al. [13] argued that the Fréchet distance is unlikely to be 3SUM-hard, because it has strongly subquadratic decision trees. However, their argument breaks down in light of a recent result showing strongly subquadratic decision trees also for 3SUM [25]. Hence, it is completely open whether the Fréchet distance is 3SUM-hard.

**Strong Exponential Time Hypothesis** The Exponential Time Hypothesis (ETH) and the Strong Exponential Time Hypothesis (SETH), both introduced by Impagliazzo, Paturi, and Zane [27, 28], provide alternative ways of proving conditional lower bounds. ETH asserts that 3-SAT has no $2^{o(N)}$ algorithm, where $N$ is the number of variables, and can be used to prove matching lower bounds for a wealth of problems, see [30] for a survey. However, since this hypothesis does not specify the exact exponent, it is not suited for proving polynomial time lower bounds, where the exponent is important.

The stronger hypothesis SETH asserts that there is no $\delta > 0$ such that $k$-SAT has an $O((2 - \delta)^N)$ algorithm for all $k$. In this paper, we will use the following weaker variant, which has also been

\(^{1}\) We always assume that $m \leq n$.

\(^{2}\) We use the term *strongly subquadratic* to differentiate between this runtime and the (mildly) subquadratic $O(n^2 \log \log n / \log n)$ algorithm from [2].
used in [33, 34].

**Hypothesis SETH’**: There is no \(O^*((2-\delta)^N)\) algorithm for CNF-SAT for any \(\delta > 0\). Here, \(O^*\) hides polynomial factors in the number of variables \(N\) and the number of clauses \(M\).

While SETH deals with formulas of width \(k\), SETH’ deals with CNF-SAT, i.e., unbounded width clauses. Thus, it is a weaker assumption and more likely to be true. Note that exhaustive search takes time \(O^*(2^N)\), and the fastest known algorithms for CNF-SAT are only slightly faster than that, namely of the form \(O^*(2^{N(1-C/\log(M/N))})\) for some positive constant \(C\) [14, 19]. Thus, SETH’ is a reasonable assumption that can be considered unlikely to fail. It has been observed that one can use SETH and SETH’ to prove lower bounds for polynomial time problems such as \(k\)-Dominating Set and others [33], the diameter of sparse graphs [34], and dynamic connectivity problems [1]. However, it seems to be applicable only for few problems, e.g., it seems to be a wide open problem to prove that 3SUM has no strongly subquadratic algorithms unless SETH fails, similarly for matching, maximum flow, edit distance, and other classic problems.

**Main result** Our main theorem gives strong evidence that the Fréchet distance may have no strongly subquadratic algorithms by relating it to the Strong Exponential Time Hypothesis.

**Theorem 1.1.** There is no \(O(n^{2-\delta})\) algorithm for the (continuous or discrete) Fréchet distance for any \(\delta > 0\), unless SETH’ fails.

Since SETH and its weaker variant SETH’ are reasonable hypotheses, by this theorem one can consider it unlikely that the Fréchet distance has strongly subquadratic algorithms. In particular, any strongly subquadratic algorithm for the Fréchet distance would not only give improved algorithms for CNF-SAT that are much faster than exhaustive search, but also for various other problems such as Hitting Set, Set Splitting, and NAE-SAT via the reductions in [18]. Alternatively, in the spirit of [33], one can view the above theorem as a possible attack on CNF-SAT, as algorithms for the Fréchet distance now could provide a route to faster CNF-SAT algorithms. In any case, anyone trying to find strongly subquadratic algorithms for the Fréchet distance should be aware that this is as hard as finding improved CNF-SAT algorithms, which might be impossible.

We remark that all our lower bounds (unless stated otherwise) hold in the Euclidean plane, and thus also in \(\mathbb{R}^d\) for any \(d \geq 2\).

**Extensions** We extend our main result in two important directions: We show approximation hardness and we prove that the lower bound still holds for restricted classes of curves.

First, it would be desirable to have good approximation algorithms in strongly subquadratic time, say a near-linear time approximation scheme. We exclude such algorithms by proving that there is no 1.001-approximation for the Fréchet distance in strongly subquadratic time unless SETH’ fails. Hence, within \(n^{o(1)}\)-factors any 1.001-approximation takes as much time as an exact algorithm.

We did not try to optimize the constant 1.001, but only to find the asymptotically largest possible approximation ratio, which seems to be a constant. We leave it as an open problem whether there is a strongly subquadratic \(O(1)\)-approximation. The literature so far contains no strongly subquadratic approximation algorithms for general curves at all.

Second, it might be conceivable that if one curve has much fewer vertices than the other, i.e., \(m \ll n\), then after some polynomial preprocessing on the smaller curve we can compute the Fréchet distance of the two curves quickly, e.g., in total time \(O((n+m^3)\log n)\). Note that such a runtime is not ruled out by the trivial argument that any algorithm needs time \(\Omega(n+m)\) for reading the
input, and is also not ruled out by Theorem 1.1, since the runtime is not subquadratic for $n = m$. We rule out such runtimes by proving that there is no $O((nm)^{1-\delta})$ algorithm “for any $m$”, unless SETH’ fails. More precisely, we prove this lower bound for the “special case” $m \approx n^\gamma$ for any constant $0 \leq \gamma \leq 1$. To make this formal, for any input parameter $\alpha$ and constants $\gamma_0 < \gamma_1$ in $\mathbb{R} \cup \{-\infty, \infty\}$, we say that a statement holds for any polynomial restriction of $n^{\gamma_0} \leq \alpha \leq n^{\gamma_1}$ if it holds restricted to instances with $n^{\gamma_0-\delta} \leq \alpha \leq n^{\gamma_1+\delta}$ for any constants $\delta > 0$ and $\gamma_0 + \delta \leq \gamma \leq \gamma_1 - \delta$. We obtain the following extension of the main result Theorem 1.1, which yields tight lower bounds for any behaviour of $m$ and any $(1 + \varepsilon)$-approximation with $0 \leq \varepsilon \leq 0.001$.

**Theorem 1.2.** There is no 1.001-approximation with runtime $O((nm)^{1-\delta})$ for the (continuous or discrete) Fréchet distance for any $\delta > 0$, unless SETH’ fails. This holds for any polynomial restriction of $1 \leq m \leq n$.

**Realistic input curves** In attempts to capture the properties of realistic input curves, strongly subquadratic algorithms have been devised for restricted classes of inputs such as backbone curves [7], $\kappa$-bounded and $\kappa$-straight [6], and $\varphi$-low density curves [21]. The most popular model are $c$-packed curves, which have been used for various generalizations of the Fréchet distance [16, 20, 26]. Driemel et al. [21] introduced this model and presented a $(1 + \varepsilon)$-approximation for the continuous Fréchet distance that runs in time $O(cn/\varepsilon + cn \log n)$, which works in any $\mathbb{R}^d$, $d \geq 2$.

While the algorithm of [21] is near-linear for small $c$ and $1/\varepsilon$, is is not clear whether its dependence on $c$ and $1/\varepsilon$ is optimal for $c$ and $1/\varepsilon$ that grow with $n$. We give strong evidence that the algorithm of [21] has optimal dependence on $c$ for any constant $0 < \varepsilon \leq 0.001$.

**Theorem 1.3.** There is no 1.001-approximation with runtime $O((cn)^{1-\delta})$ for the (continuous or discrete) Fréchet distance on $c$-packed curves for any $\delta > 0$, unless SETH’ fails. This holds for any polynomial restriction of $1 \leq c \leq n$.

Since we prove this claim for any polynomial restriction $c \approx n^\gamma$, the above result excludes 1.001-approximations with runtime, say, $O(c^2 + n)$.

Regarding the dependence on $\varepsilon$, in any dimension $d \geq 5$ we can prove a conditional lower bound that matches the dependency on $\varepsilon$ of [21] up to a polynomial.

**Theorem 1.4.** There is no $(1 + \varepsilon)$-approximation for the (continuous or discrete) Fréchet distance on $c$-packed curves in $\mathbb{R}^d$, $d \geq 5$, with runtime $O(\min\{cn/\sqrt{\varepsilon}, n^2\})^{1-\delta}$ for any $\delta > 0$, unless SETH’ fails. This holds for sufficiently small $\varepsilon > 0$ and any polynomial restriction of $1 \leq c \leq n$ and $\varepsilon \leq 1$.

**Outline of the main result** To prove the main result we present a reduction from CNF-SAT to the Fréchet distance. Given a CNF-SAT instance $\varphi$, we partition its variables into sets $V_1, V_2$ of equal size. In order to find a satisfying assignment of $\varphi$ we have to choose (partial) assignments $a_1$ of $V_1$ and $a_2$ of $V_2$. We will construct curves $P_1, P_2$ where $P_k$ is responsible for choosing $a_k$. To this end, $P_k$ consists of assignment gadgets, one for each assignment of $V_k$. Assignment gadgets are built of clause gadgets, one for each clause. The assignment gadgets of assignments $a_1$ of $V_1$ and $a_2$ of $V_2$ are constructed such that they have Fréchet distance at most 1 if and only if $(a_1, a_2)$ forms a satisfying assignment of $\varphi$. In $P_1$ and $P_2$ we connect these assignment gadgets with some additional curves to implement an OR-gadget, which forces any traversal of $(P_1, P_2)$ to walk along two assignment gadgets in parallel. If $\varphi$ is not satisfiable, then any pair of assignment gadgets has Fréchet distance larger than 1, so that $P_1, P_2$ have Fréchet distance larger than 1. If, on the other hand, a satisfying assignment $(a_1, a_2)$ of $\varphi$ exists, then we ensure that there is a traversal of $P_1, P_2$.
that essentially only traverses the assignment gadgets of $a_1$ and $a_2$ in parallel, so that it always stays in distance 1.

To argue about the runtime, since $P_k$ contains an assignment gadget for every assignment of one half of the variables, and every assignment gadget has polynomial size in $M$, there are $n = \mathcal{O}^*(2^{N/2})$ vertices on each curve. Thus, any $\mathcal{O}(n^{2-\delta})$ algorithm for the Fréchet distance would yield an $\mathcal{O}^*(2^{(1-\delta/2)N})$ algorithm for CNF-SAT, contradicting SETH'.

**Remark: Orthogonal Vectors** Let Orthog be the problem of ‘finding a pair of orthogonal vectors”: given two sets $S_1, S_2 \subseteq \{0,1\}^d$ of $n$ vectors each, determine if there are $u \in S_1$ and $v \in S_2$ with $\langle u, v \rangle = \sum_{i=1}^d u_i v_i = 0$, where the sum is computed over the integers, see [35, 36]. Clearly, Orthog can be solved in time $\mathcal{O}(n^2d)$. However, Orthog has no strongly subquadratic algorithms unless SETH' fails. More precisely, in [35] it was shown that SETH' implies the following statement.

OrthogHypothesis: There is no algorithm for Orthog with runtime $\mathcal{O}(n^{2-\delta}d^{O(1)})$ for any $\delta > 0$.

All known conditional lower bounds based on SETH' implicitly go through Orthog or some variant of this problem. In fact, this is also the case for our results, as is easily seen by going through the proof in [35] and noting that we use the same tricks. Specifically, given a CNF-SAT instance $\phi$ on variables $x_1, \ldots, x_N$ and clauses $C_1, \ldots, C_M$, we split the variables into two halves $V_1, V_2$ of equal size and enumerate all assignments $A_k$ of true and false to $V_k$. Then every clause $C_i$ specifies sets $B_k^i \subseteq A_k$ of partial assignments that do not make $C_i$ become true. Clearly, a satisfying assignment $(a_1, a_2) \in A_1 \times A_2$ has to evade $B_1^i \times B_2^i$ for all $i$. This problem is equivalent to an instance of Orthog with $d = M$ and $n = 2^{N/2}$, where $S_k$ contains a vector for every partial assignment $a_k \in A_k$ and the $i$-th position of this vector is 1 or 0, depending on whether $a_k \in B_k^i$ or not. In our proof, we could replace this instance by an arbitrary instance of Orthog, yielding a reduction from Orthog to the Fréchet distance.

Hence, in Theorems 1.1, 1.3, and 1.4 we could replace the assumption “unless SETH' fails” by the weaker assumption “unless OrthogHypothesis fails”. This is a stronger statement, since there is only more reason to believe that Orthog has no strongly subquadratic algorithms than that there is for believing that CNF-SAT takes time $2^{N-o(N)}$. Moreover, it shows a relation between two polynomial time problems, Orthog and the Fréchet distance.

For Theorem 1.2 we would need an imbalanced version of the OrthogHypothesis, where the two sets $S_1, S_2$ have different sizes $n_1, n_2$. Then unless SETH' fails there is no $\mathcal{O}(n_1n_2^{1-\delta}d^{O(1)})$ algorithm for any $\delta > 0$, and this holds for any polynomial restriction of $1 \leq n_1 \leq n_2$, which follows from a slight generalization of [35]. If we state this implication of SETH' as a hypothesis OrthogHypothesis*, then in Theorem 1.2 we could replace “unless SETH' fails” by the weaker assumption “unless OrthogHypothesis* fails”.

**Organization** We start by defining the variants of the Fréchet distance, $c$-packedness, and other basic notions in Section 2. Section 3 deals with general curves. We prove the main result for the discrete Fréchet distance in less than 3 pages in Section 3.1. This construction also already proves inapproximability. We generalize the proof to the continuous Fréchet distance in Section 3.2 (which is more tedious than in the discrete case) and to $m \ll n$ in Section 3.3 (which is an easy trick). Section 4 deals with $c$-packed curves. In Section 4.1 we present a new OR-gadget that generates less packed curves; plugging in the curves constructed in the main result proves Theorem 1.3. In Section 4.2 we make use of the fact that in $\geq 4$ dimensions there are point sets $Q_1, Q_2$ of arbitrary
size with each pair of points \((q_1, q_2)\) having distance exactly 1. This allows to construct less packed curves that we plug into the OR-gadget from the preceding section to prove Theorem 1.4.

2 Preliminaries

For \(N \in \mathbb{N}\) we let \([N] := \{1, \ldots, N\}\). A (polygonal) curve \(P\) is defined by its vertices \(p_1, \ldots, p_n\). We view \(P\) as a continuous function \(P : [0, n] \to \mathbb{R}^d\) with \(P(i + \lambda) = (1 - \lambda)p_i + \lambda p_{i+1}\) for \(i \in [n - 1]\), \(\lambda \in [0, 1]\). We write \(|P| = n\) for the number of vertices of \(P\). For two curves \(P_1, P_2\) we let \(P_1 \circ P_2\) be the curve on \(|P_1| + |P_2|\) vertices that first follows \(P_1\), then walks along the segment from \(P_1(\lfloor P_1\rfloor)\) to \(P_2(0)\), and then follows \(P_2\). In particular, for two points \(p, q \in \mathbb{R}^d\) the curve \(p \circ q\) is the segment from \(p\) to \(q\), and any curve \(P\) on vertices \(p_1, \ldots, p_n\) can be written as \(P = p_1 \circ \ldots \circ p_n\).

Consider a curve \(P\) and two points \(p_1 = P(\lambda_1), p_2 = P(\lambda_2)\) with \(\lambda_1, \lambda_2 \in [0, n]\). We say that \(p_1\) is within distance \(D\) of \(p_2\) along \(P\) if the length of the subcurve of \(P\) between \(P(\lambda_1)\) and \(P(\lambda_2)\) is at most \(D\).

**Variants of the Fréchet distance**  Let \(\Phi_n\) be the set of all continuous and non-decreasing functions \(\phi\) from \([0, 1]\) onto \([0, n]\). The continuous Fréchet distance between two curves \(P_1, P_2\) with \(|P_1| = n, |P_2| = m\) is defined as

\[
d_F(P_1, P_2) := \inf_{\phi_1 \in \Phi_n} \max_{t \in [0, 1]} \|P_1(\phi_1(t)) - P_2(\phi_2(t))\|
\]

where \(\|\cdot\|\) denotes the Euclidean distance. We call \((\phi_1, \phi_2)\) a (continuous) traversal of \((P_1, P_2)\), and say that it has width \(D\) if \(\max_{t \in [0, 1]} \|P_1(\phi_1(t)) - P_2(\phi_2(t))\| \leq D\).

In the discrete case, we let \(\Delta_n\) be the set of all non-decreasing functions \(\phi\) from \([0, 1]\) onto \([n]\). The discrete Fréchet distance between two curves \(P_1, P_2\) with \(|P_1| = n, |P_2| = m\) is then defined as

\[
d_{dF}(P_1, P_2) := \inf_{\phi_1 \in \Delta_n} \max_{t \in [0, 1]} \|P_1(\phi_1(t)) - P_2(\phi_2(t))\|
\]

We obtain an analogous notion of a (discrete) traversal and its width. Note that any \(\phi \in \Delta_n\) is a staircase function attaining all values in \([n]\). Hence, \((\phi_1(t), \phi_2(t))\) changes only at finitely many points in time \(t\). At any such time step we jump to the next vertex in \(P_1\) or \(P_2\) or both.

It is known that for any curves \(P_1, P_2\) we have \(d_F(P_1, P_2) \leq d_{dF}(P_1, P_2)\) [22].

**Realistic input curves**  As an example of input restrictions that resemble practical input curves we consider the model of [21]. A curve \(P\) is \(c\)-packed if for any point \(q \in \mathbb{R}^d\) and any radius \(r > 0\) the total length of \(P\) inside the ball \(B(q, r)\) is at most \(cr\). Here, \(B(q, r)\) is the ball of radius \(r\) around \(q\). In this paper, we say that a curve \(P\) is \(\Theta(c)\)-packed, if there are constants \(\alpha > \beta > 0\) such that \(P\) is \(\alpha c\)-packed but not \(\beta c\)-packed.

This model is well motivated from a practical point of view. Examples of classes of \(c\)-packed curves are boundaries of convex polygons and \(\gamma\)-fat shapes as well as algebraic curves of bounded maximal degree (see [21]).

**Satisfiability**  In CNF-SAT we are given a formula \(\varphi\) on variables \(x_1, \ldots, x_N\) and clauses \(C_1, \ldots, C_M\) in conjunctive normal form with unbounded clause width. Let \(V\) be any subset of the variables of \(\varphi\). Let \(a\) be any assignment of \(T\) (true) or \(F\) (false) to the variables of \(V\). We call \(a\) a partial assignment and say that \(a\) satisfies a clause \(C = \bigvee_{i \in I} x_i \lor \bigvee_{j \in J} \neg x_i\) if for some \(i \in I \cap V\)
we have \( a(x_i) = T \) or for some \( i \in J \cap V \) we have \( a(x_i) = F \). We denote by \( \text{sat}(a, C) \) whether partial assignment \( a \) satisfies clause \( C \). Note that assignments \( a \) of \( V \) and \( a' \) of the remaining variables \( V' \) form a satisfying assignment \((a, a')\) of \( \varphi \) if and only if we have \( \text{sat}(a, C_i) \lor \text{sat}(a', C_i) = T \) for all \( i \in \{1, \ldots, M\} \).

All bounds that we prove in this paper assume the hypothesis SETH' (see Section 1), which asserts that \( \text{CNF-SAT} \) has no \( O^*((2-\delta)^N) \) algorithm for any \( \delta > 0 \). Here, \( O^* \) hides polynomials factors in \( N \) and \( M \). The following is an easy corollary of SETH'.

**Lemma 2.1.** There is no \( O^*((2-\delta)^N) \) algorithm for \( \text{CNF-SAT} \) restricted to formulas with \( N \) variables and \( M \leq 2^{\delta N} \) clauses for any \( \delta, \delta' > 0 \), unless SETH' fails.

**Proof.** Any such algorithm would imply an \( O^*((2-\delta)^N) \) algorithm for \( \text{CNF-SAT} \) (with no restrictions on the input), since for \( M \leq 2^{\delta N} \) we can run the given algorithm, while for \( M > 2^{\delta N} \) we can decide satisfiability in time \( O(M2^{N}) = O(M^{1+1/\delta'}) = O^*(1) \). \( \square \)

### 3 General curves

We first present a reduction from \( \text{CNF-SAT} \) to the Fréchet distance and show that it proves Theorem 1.1 for the discrete Fréchet distance. In Section 3.2 we then show that the same construction also works for the continuous Fréchet distance. Finally, in Section 3.3 we generalize these results to curves with imbalanced numbers of vertices \( n, m \) to show Theorem 1.2.

#### 3.1 The basic reduction, discrete case

Let \( \varphi \) be a given \( \text{CNF-SAT} \) instance with variables \( x_1, \ldots, x_N \) and clauses \( C_1, \ldots, C_M \). We split the variables into two halves \( V_1 := \{x_1, \ldots, x_{N/2}\} \) and \( V_2 := \{x_{N/2+1}, \ldots, x_N\} \). For \( k \in \{1, 2\} \) let \( A_k \) be all assignments\(^4\) of \( T \) or \( F \) to the variables in \( V_k \), so that \( |A_k| = 2^{N/2} \). In the whole section we let \( \varepsilon := 1/1000 \).

We will construct two curves \( P_1, P_2 \) such that \( d_{AF}(P_1, P_2) \leq 1 \) if and only if \( \varphi \) is satisfiable. In the construction we will use gadgets as follows.

**Clause gadgets**  This gadget encodes whether a partial assignment satisfies a clause. We set for \( i \in \{0, 1\} \)

\[
\begin{align*}
     c_{1,T}^i &:= (i/3, \frac{1}{2} - \varepsilon), & c_{1,F}^i &:= (i/3, \frac{1}{2} + \varepsilon), \\
     c_{2,T}^i &:= (i/3, -\frac{1}{2} + \varepsilon), & c_{2,F}^i &:= (i/3, -\frac{1}{2} - \varepsilon).
\end{align*}
\]

Let \( k \in \{1, 2\} \). For any partial assignment \( a_k \in A_k \) and clause \( C_i \), \( i \in [M] \), we construct a clause gadget consisting of a single point,

\[
CG(a_k, i) := c_{k,\text{sat}(a_k, C_i)}^{i \mod 2}.
\]

Thus, if assignment \( a_k \) satisfies clause \( C_i \) then the corresponding clause gadget is nearer to the clause gadgets associated with \( A_{3-k} \). Explicitly calculating all pairwise distances of these points, we obtain the following lemma.

**Lemma 3.1.** Let \( a_k \in A_k \), \( k \in \{1, 2\} \), and \( i, j \in [M] \). If \( i \equiv j \mod 2 \) and \( \text{sat}(a_1, C_i) \lor \text{sat}(a_2, C_j) = T \) then \( \|CG(a_1, i) - CG(a_2, j)\| \leq 1 \). Otherwise \( \|CG(a_1, i) - CG(a_2, j)\| \geq 1 + 2\varepsilon \).

\(^4\)In later sections we will replace \( V_1, V_2 \) by different partitionings and \( A_1, A_2 \) by subsets of all assignments. The lemmas in this section are proven in a generality that allows this extension.
Assignment gadgets  This gadget consists of clause gadgets and encodes the set of satisfied clauses for an assignment. We set
\[ r_1 := \left( -\frac{1}{3}, \frac{1}{2} \right), \quad r_2 := \left( -\frac{1}{3}, -\frac{1}{2} \right). \]
The assignment gadget for any \( a_k \in A_k \) consists of the starting point \( r_2 \) followed by all clause gadgets of \( a_k \),
\[ AG(a_k) := r_k \circ \bigcap_{i \in [M]} CG(a_k, i), \]
(recall the definition of \( \circ \) in Section 2). The figure to the right shows an assignment gadget on \( M = 2 \) clauses at the top and an assignment gadget on \( M = 4 \) clauses at the bottom. The arrows indicate the order in which the segments are traversed.

**Lemma 3.2.** Let \( a_k \in A_k, k \in \{1, 2\} \). If \((a_1, a_2)\) is a satisfying assignment of \( \varphi \) then \( d_{DF}(AG(a_1), AG(a_2)) < 1 \). If \((a_1, a_2)\) is not satisfying then \( d_{DF}(AG(a_1), AG(a_2)) > 1 + \varepsilon \), and we even have \( d_{DF}(AG(a_1) \circ \pi_1, AG(a_2) \circ \pi_2) > 1 + \varepsilon \) for any curves \( \pi_1, \pi_2 \).

*Proof.* If \((a_1, a_2)\) is satisfying then the parallel traversal
\[ (r_1, r_2), (CG(a_1, 1), CG(a_2, 1)), \ldots, (CG(a_1, M), CG(a_2, M)) \]
has width 1 by Lemma 3.1.

Assume for the sake of contradiction that \((a_1, a_2)\) is not satisfying but there is a traversal of \((AG(a_1) \circ \pi_1, AG(a_2) \circ \pi_2)\) with width \( 1 + \varepsilon \). Observe that \( ||r_1 - r_2|| = 1 \) and \( ||r_k - c_{3-k,x}|| \geq 1 + 2\varepsilon \) for any \( k \in \{1, 2\}, i \in \{0, 1\}, x \in \{T, F\} \). Thus, the traversal has to start at positions \((r_1, r_2)\) and then step to positions \((CG(a_1, 1), CG(a_2, 1))\), as advancing in only one of the curves leaves us in distance larger than \( 1 + \varepsilon \). Inductively and using Lemma 3.1, the same argument shows that in the \( i \)-th step we are at positions \((CG(a_1, i), CG(a_2, i))\) for any \( i \in [M] \). Since there is an unsatisfied clause \( C_i \), so that \( ||CG(a_1, i) - CG(a_2, i)|| > 1 + 2\varepsilon \) by Lemma 3.1, we obtain a contradiction. \( \square \)

**Construction of the curves** The curve \( P_k \) will consist of all assignment gadgets for assignments \( A_k, k \in \{1, 2\} \), plus some additional points. The additional points implement an OR-gadget over the assignment gadgets, by enforcing that any traversal of \((P_1, P_2)\) with width \( 1 + \varepsilon \) has to traverse two assignment gadgets in parallel, and traversing one pair of assignment gadgets in parallel suffices.

We define the following control points,
\[
\begin{align*}
s_1 &= \left( -\frac{1}{3}, \frac{1}{5} \right), \quad t_1 = \left( \frac{1}{3}, \frac{1}{5} \right), \\
s_2 &= \left( -\frac{1}{3}, 0 \right), \quad t_2 = \left( \frac{1}{3}, 0 \right), \quad s^*_2 = \left( -\frac{1}{3}, -\frac{4}{5} \right), \quad t^*_2 = \left( \frac{1}{3}, -\frac{4}{5} \right).
\end{align*}
\]

Finally, we set
\[
\begin{align*}
P_1 &= \bigcap_{a_1 \in A_1} (s_1 \circ AG(a_1) \circ t_1), \\
P_2 &= s_2 \circ s^*_2 \circ \left( \bigcap_{a_2 \in A_2} AG(a_2) \right) \circ t^*_2 \circ t_2.
\end{align*}
\]

The figure to the right shows \( P_1 \) (dotted) and \( P_2 \) (solid) in an example with \( M = 2 \) clauses and unrealistically only two assignments.

Let \( Q_k \) be the vertices that may appear in \( P_k \), i.e., \( Q_1 = \{s_1, t_1, r_1, c^0_{1,F}, c^0_{1,T}, c^1_{1,F}, c^1_{1,T}\} \) and \( Q_2 = \{s_2, t_2, r_2, s^*_2, t^*_2, \hat{c}^0_{1,F}, \hat{c}^0_{1,T}, \hat{c}^1_{1,F}, \hat{c}^1_{1,T}\} \). Explicitly calculating all pairwise distances of all points, we obtain the following lemma.
Lemma 3.3. No pair \((q_1,q_2) \in Q_1 \times Q_2\) has \(\|q_1 - q_2\| \in (1,1+\varepsilon)\). Moreover, the set \(\{(q_1,q_2) \in Q_1 \times Q_2 \mid \|q_1 - q_2\| \leq 1\}\) consists of the following pairs:

\[
(q,s_2), (q,t_2) \text{ for any } q \in Q_1,
(s_1,q) \text{ for any } q \in Q_2 \setminus \{t_2\},
(t_1,q) \text{ for any } q \in Q_2 \setminus \{s_2\},
(r_1,r_2),
(c^1_{i,x},c^1_{i,y}) \text{ for } x \lor y = T \text{ where } i \in \{0,1\}, x,y \in \{T,F\}.
\]

Correctness We show that if \(\varphi\) is satisfiable then \(d_{df}(P_1,P_2) \leq 1\), while otherwise \(d_{df}(P_1,P_2) > 1 + \varepsilon\).

Lemma 3.4. If \(d_{df}(P_1,P_2) \leq 1 + \varepsilon\) then \(A_1 \times A_2\) contains a satisfying assignment.

Proof. By Lemma 3.3 any traversal with width \(1 + \varepsilon\) also has width 1. Consider any traversal of \((P_1,P_2)\) with width 1. Consider any time step \(T\) at which we are at position \(s^*_2\) in \(P_2\). The only point in \(P_1\) that is within distance 1 of \(s^*_2\) is \(s_1\), say we are at the copy of \(s_1\) that comes right before assignment gadget \(AG(a_1)\), \(a_1 \in A_1\). Following time step \(T\), we have to start traversing \(AG(a_1)\), so consider the first time step \(T'\) where we are at the point \(r_1\) in \(AG(a_1)\). The only points in \(P_2\) within distance 1 of \(r_1\) are \(s_2, t_2,\) and \(r_2\). Note that we already passed \(s^*_2\) in \(P_2\) by time \(T\), so we cannot be in \(s_2\) at time \(T'\). Moreover, in between \(T\) and \(T'\) we are only at \(s_1\) and \(r_1\) in \(P_1\), which have distance larger than 1 to \(t_2^*\). Thus, we cannot pass \(t_2^*\), and we cannot be at \(t_2\) at time \(T'\). Hence, we are at \(r_2\), say at the copy of \(r_2\) in assignment gadget \(AG(a_2)\) for some \(a_2 \in A_2\). The yet untraversed remainder of \(P_k\) is of the form \(AG(a_k) \circ \pi_k\) for \(k \in \{1,2\}\). Since our traversal of \((P_1,P_2)\) has width 1, we obtain \(d_{df}(AG(a_1) \circ \pi_1, AG(a_2) \circ \pi_2) \leq 1\). By Lemma 3.2, \((a_1,a_2)\) forms a satisfying assignment of \(\varphi\).

Lemma 3.5. If \(A_1 \times A_2\) contains a satisfying assignment then \(d_{df}(P_1,P_2) \leq 1\).

Proof. Let \((a_1,a_2) \in A_1 \times A_2\) be a satisfying assignment of \(\varphi\). We describe a traversal through \(P_1, P_2\) with width 1. We start at \(s_2 \in P_2\) and the first point of \(P_1\). We stay at \(s_2\) and follow \(P_1\) until we arrive at the copy of \(s_1\) that comes right before \(AG(a_1)\) (note that \(s_2\) has distance 1 to any point in \(P_1\)). Then we stay at \(s_1\) and follow \(P_2\) until we arrive at the copy of \(r_2\) in \(AG(a_2)\) (note that the only point that is too far away from \(s_1\) is \(t_2^*\), but this point comes after all assignment gadgets in \(P_2\)). In the next step we go to positions \((r_1,r_2)\) (in \(AG(a_1), AG(a_2)\)). Then we follow the clause gadgets \((CG(a_1,i), CG(a_2,i))\) in parallel, always staying within distance 1 by Lemma 3.1. In the next step we stay at \(CG(a_2,M)\) and go to \(t_1\) in \(P_1\) (which has distance 1 to any point in \(P_2\) except for \(s^*_2\), which we will never encounter again). We stay at \(t_1\) in \(P_1\) and follow \(P_2\) completely until we arrive at its endpoint \(t_2\). Since \(t_2\) has distance 1 to any point in \(P_1\), we can now stay at \(t_2\) in \(P_2\) and follow \(P_1\) to its end.

Proof of Theorem 1.1, discrete case Note that we have

\[
n = \max\{|P_1|,|P_2|\} = O(M) \cdot \max\{|A_1|,|A_2|\} = O(M \cdot 2^{N/2}).
\]

Moreover, the instance \((P_1,P_2)\) can be constructed in time \(O(NM2^{N/2})\). Any \((1+\varepsilon)\)-approximation can decide whether \(d_{df}(P_1,P_2) \leq 1\) or \(d_{df}(P_1,P_2) > 1 + \varepsilon\), which by Lemmas 3.4 and 3.5 yields an algorithm that decides whether \(\varphi\) is satisfiable. If such an algorithm runs in time \(O(n^{2-\delta})\) for any small \(\delta > 0\), then the resulting \(\text{CNF}-\text{SAT}\) algorithm runs in time \(O(M^{2(1-\delta/2)N})\), contradicting SETH'.
3.2 Continuous case

The construction from the last section also works for the continuous Fréchet distance. However, for unsatisfiable formulas it becomes tedious to argue that continuous traversals are not much better than discrete traversals. For instance, we have to argue that we cannot stay at a fixed point between the clause gadgets $e^0_{1,T}$ and $e^1_{1,T}$ while traversing more than one clause gadget in $P_2$.

We adapt the proof from the last section on the same curves $P_1, P_2$ to work for the continuous Fréchet distance. To this end, we have to reprove Lemmas 3.4 and 3.5. We will make use of the following property. Here, we set $\text{sym}(\text{CG}(a_1, i)) := \text{CG}(a_2, i)$ and $\text{sym}(r_1) := r_2$ and interpolate linearly between them to obtain a symmetric point in $AG(a_2)$ for every point in $AG(a_1)$ (for any fixed $a_1 \in A_1, a_2 \in A_2$). We also set $\text{sym}(\text{sym}(p_1)) := p_1$, to obtain a symmetric point in $AG(a_1)$ for every point in $AG(a_2)$.

**Lemma 3.6.** Consider any points $p_k$ in $AG(a_k)$, $k \in \{1, 2\}$, with $\|p_1 - p_2\| \leq 1 + \varepsilon$. Then we have $\|p_2 - \text{sym}(p_1)\| \leq \frac{1}{10}$ and $\|\text{sym}(p_2) - p_1\| \leq \frac{1}{10}$.

**Proof.** Let $p_k = (x_k, y_k)$ and note that we have $|y_1 - y_2| \geq 1 - 2\varepsilon$. Thus, if $|x_1 - x_2| > \frac{1}{9} - 2\varepsilon$ then we have (recall that $\varepsilon = 1/1000$)

$$\|p_1 - p_2\| > \sqrt{(\frac{1}{9} - 2\varepsilon)^2 + (1 - 2\varepsilon)^2} > 1 + \varepsilon,$$

a contradiction. Since $\text{sym}(p_1) = (x_1, y'_1)$ with $|y'_1 - y_2| \leq 2\varepsilon$, we obtain

$$\|p_2 - \text{sym}(p_1)\| \leq \sqrt{(\frac{1}{9} - 2\varepsilon)^2 + (2\varepsilon)^2} \leq \frac{1}{10}.$$ 

and the same bound holds for $\|\text{sym}(p_2) - p_1\|$. \qed

**Lemma 3.7.** (Analogue of Lemma 3.4) If $d_F(P_1, P_2) \leq 1 + \varepsilon = 1.001$ then $A_1 \times A_2$ contains a satisfying assignment.

**Proof.** In this proof, we say that two points $p_1 = (x_1, y_1)$, $p_2 = (x_2, y_2)$ have $y$-distance $D$ if $|y_1 - y_2| \leq D$.

Consider any traversal of $(P_1, P_2)$ with width $1 + \varepsilon$. Consider any time step $T$ where we are at position $s_2^2$ in $P_2$. The only points in $P_1$ that are within distance $1 + \varepsilon$ of $s_2^2$ are within distance $1/20$ and $y$-distance $\varepsilon$ of $s_1$ (since no point in $P_1$ has lower $y$-value than $s_1$ and $\sqrt{1 + (1/20)^2} > 1+\varepsilon$). Say we are near the copy of $s_1$ that comes right before assignment gadget $AG(a_1)$, $a_1 \in A_1$. Following time step $T$, we have to start traversing $AG(a_1)$, so consider the first time step $T'$ where we are at the point $r_1$ in $AG(a_1)$. The only points in $P_2$ within distance $1 + \varepsilon$ of $r_1$ are near $s_2, t_2$, or $r_2$. Note that we already passed $s_2^2$ in $P_2$ by time $T$, so we cannot be near $s_2$ at time $T'$. Moreover, in between $T$ and $T'$ we are always near $s_1$ or between $s_1$ and $r_1$ in $P_1$, so we are always above and to the left of $s_1 + (1/20, 0)$, which has distance larger than $1 + \varepsilon$ to $t_2^*$. Thus, we cannot pass $t_2^*$, and we cannot be near $t_2$ at time $T'$. Hence, we are near $r_2$, more precisely, we are in distance $1/20$ and $y$-distance $\varepsilon$ of $r_2$ (this is the same situation as for $s_1$ and $s_2^2$). After that, the traversal has to further traverse $AG(a_1)$ and/or $AG(a_2)$. Consider the first time step at which we are at $CG(a_1, 1)$ or $CG(a_2, 1)$, say we reach $CG(a_1, 1)$ first. By Lemma 3.6, we are within distance $1/9$ of $CG(a_1, 1)$. Since we were near $r_2$ at time $T'$, we now passed $r_2$, and since we did not pass $CG(a_2, 1)$ yet, we are even within distance $1/9$ of $CG(a_2, 1)$ along the curve $P_2$. This proves the induction base of the following inductive claim.

**Claim 3.8.** Let $T_i$ be the first step in time at which the traversal is at $CG(a_1, i)$ or $CG(a_2, i)$, $i \in [M]$. At time $T_i$ the traversal is within distance $1/9$ of $CG(a_k, i)$ along the curve $P_k$ for both $k \in \{1, 2\}$. 

10
Proof. Note that at all times $T_i$ (and in between) Lemma 3.6 is applicable, so we clearly are within distance $1/9$ of $CG(a_i,i+1)$ at time $T_{i+1}$ for any $i \in [M]$, $k \in \{1,2\}$. Since $\|CG(a_k,i)-CG(a_k,i+1)\| \geq 1/3$, points within distance $1/9$ of $CG(a_k,i)$ are not within distance $1/9$ of $CG(a_k,i+1)$. Hence, if we are within distance $1/9$ of $CG(a_k,i)$ along $P_k$ for both $k \in \{1,2\}$ at time $T_i$, then at time $T_{i+1}$ we passed $CG(a_k,i)$ and did not pass $CG(a_k,i+1)$ (yet (by definition of $T_{i+1}$), so that we are within distance $1/9$ of $CG(a_k,i+1)$ along $P_k$ for both $k \in \{1,2\}$. \hfill \Box

Finally, we show that the above claim implies that $(a_1,a_2)$ is a satisfying assignment. Assume for the sake of contradiction that some clause $C_i$ is not satisfied by both $a_1$ and $a_2$. Say at time $T_i$ we are at $CG(a_1,i)$ (if we are at $CG(a_2,i)$ instead, then a symmetric argument works). At the same time we are at some point $p$ in $AG(a_2)$. By the above claim, $p$ is within distance $1/9$ of $CG(a_2,i)$ along $P_2$. Note that $p$ lies on any of the line segments $c_{2,T}^0 \circ c_{1,F}^1$, $c_{2,F}^0 \circ c_{2,T}^1$, $c_{2,F}^0 \circ c_{2,F}^1$, or $r_2 \circ c_{2,F}^0$, since $\text{sat}(a_2,C_i) = F$. In any case, the current distance $\|p- CG(a_1,i)\|$ is at least the distance from the point $c_{1,F}^0$ to the line through $c_{2,F}^0$ and $c_{2,T}^1$. We compute this distance as

$$\frac{1}{3}(1+\varepsilon) \sqrt{\frac{(1/3)^2 + (2\varepsilon)^2}{}} > 1 + \varepsilon,$$

which contradicts the traversal having width $1 + \varepsilon$. \hfill \Box

Lemma 3.9. (Analogue of Lemma 3.5) If $A_1 \times A_2$ contains a satisfying assignment then $d_F(P_1,P_2) \leq 1$.

Proof. Follows from Lemma 3.5 and the general inequality $d_F(P_1,P_2) \leq d_{af}(P_1,P_2)$. \hfill \Box

3.3 Generalization to imbalanced numbers of vertices

Assume that the input curves $P_1,P_2$ have different numbers of vertices $n = |P_1|$, $m = |P_2|$ with $n \geq m$. We show that there is no $O((nm)^{1-\delta})$ algorithm for the Fréchet distance for any $\delta > 0$, even for any polynomial restriction of $1 \leq m \leq n$. More precisely, for any $\delta \leq \gamma \leq 1 - \delta$ we show that there is no $O((nm)^{1-\delta})$ algorithm for the Fréchet distance restricted to instances with $n^{\gamma - \delta} \leq m \leq n^{\gamma + \delta}$.  

To this end, given a CNF-SAT instance $\varphi$ we partition its variables $x_1, \ldots, x_N$ into $4$ $V_k := \{x_1, \ldots, x_{\ell_k}\}$ and $V_k' := \{x_{\ell_k+1}, \ldots, x_N\}$ and let $A_k'$ be all assignments of $V_k'$, $k \in \{1,2\}$. Note that $|A_k'| = 2^{|V_k'|} = 2^\ell$ and $|A_k'| = 2^{N-\ell}$. Now we use the same construction as in Section 3.1 but replace $V_k$ by $V_k'$ and $A_k$ by $A_k'$. Again we obtain that any 1.001-approximation for the Fréchet distance of the constructed curves $P_1,P_2$ decides satisfiability of $\varphi$. Observe that the constructed curves contain a number of points of

$$n = |P_1| = \Theta(M \cdot |A_1'|), \; m = |P_2| = \Theta(M \cdot |A_2'|).$$

Hence, any 1.001-approximation of the Fréchet distance with runtime $O(((nm)^{1-\delta})$ for any small $\delta > 0$ yields an algorithm for CNF-SAT with runtime $O(M^2(2^{\ell}2^N2^\ell))=O(M^2(2^{(1-\delta)N}))$, contradicting SETH'.

Finally, we set $\ell := N/(\gamma + 1)$ (rounded in any way) so that $|A_1'| = \Theta(2^{N/(\gamma + 1)})$ and $|A_2'| = \Theta(2^{N\gamma/(\gamma + 1)})$. Using Lemma 2.1 we can assume that $1 \leq M \leq 2^{N/4}$. Hence, we have

$$\Omega(2^{N/(\gamma + 1)}) \leq n \leq \Omega(2^{N/(\gamma + 1)+\delta N/4}),$$

$$\Omega(2^{N\gamma/(\gamma + 1)}) \leq m \leq \Omega(2^{N\gamma/(\gamma + 1)+\delta N/4}),$$

For the impatient reader: we will set $\ell := N/(\gamma + 1)$ (rounded in any way).
which implies $\Omega(n^{\gamma-\delta/2}) \leq m \leq \mathcal{O}(n^{\gamma+\delta/2})$. For sufficiently large $n$, we obtain the desired polynomial restriction $n^{\gamma-\delta} \leq m \leq n^{\gamma+\delta}$. This proves Theorem 1.2.

## 4 Realistic inputs: $c$-packed curves

### 4.1 Constant factor approximations

The curves constructed in Section 3.1 are highly packed, since all assignment gadgets lie roughly in the same area. Specifically they are not $o(n)$-packed. In this section we want to construct $c$-packed instances and show that there is no $1.001$-approximation with runtime $\mathcal{O}((cn)^{1-\delta})$ for any $\delta > 0$ for the Fréchet distance unless SETH fails, not even restricted to instances with $n^{\gamma-\delta} \leq c \leq n^{\gamma+\delta}$ for any $\delta \leq \gamma \leq 1 - \delta$. This proves Theorem 1.3.

To this end, we again consider a CNF-SAT instance $\varphi$, partition its variables $x_1, \ldots, x_N$ into two sets $V_1, V_2$ of size $N/2$, and consider the set $A_k$ of all assignments of $T$ and $F$ to the variables in $V_k$. Now we partition $A_k$ into sets $A_{k1}, \ldots, A_{k\ell}$ of size $\Theta(2^{N/2}/\ell)$, where we fix $1 \leq \ell \leq 2^{N/2}$ later. Formula $\varphi$ is satisfiable if and only if for some pair $(j_1, j_2) \in [\ell]^2$ the set $A_{j_1} \times A_{j_2}$ contains a satisfying assignment. This suggests to use the construction of Section 3.1 after replacing $A_1$ by $A_{j_1}$ and $A_2$ by $A_{j_2}$, yielding a pair of curves $(P_{1j_1j_2}, P_{2j_1j_2})$. Now, $\varphi$ is satisfiable if and only if $d_F(P_{1j_1j_2}, P_{2j_1j_2}) \leq 1$ for some $(j_1, j_2) \in [\ell]^2$. For the sake of readability, we rename the constructed curves slightly so that we have curves $(P_1, P_2)$ for $j \in [\ell]$.

**OR-gadget** In the whole section we let $\rho := 1/\sqrt{2}$. We present an OR-construction over the gadgets $(P_1, P_2)$ that is not too packed, in contrast to the OR-construction over assignment gadgets that we used in Section 3.1. We start with two building blocks, where for any $j \in \mathbb{N}$ we set

\[
U_L(j) := ((j+1)\rho, \rho) \circ ((j+1)\rho, 2\rho) \circ ((j+1)\rho, \rho),
\]

\[
U_R(j) := ((j+1)\rho, \rho) \circ ((j+1)\rho, 3\rho) \circ ((j+1)\rho, \rho) \circ (j\rho, 0).
\]

Moreover, we set $U(j) := U_L(j) \circ U_R(j)$. For a curve $\pi$ and $z \in \mathbb{R}$ we let $\operatorname{tr}_z(\pi)$ be the curve $\pi$ translated by $z$ in $x$-direction. The OR-gadget now consists of the following two curves,

\[
R_1 := \bigcirc_{j=1}^{\ell^2} (U_L(2j) \circ \operatorname{tr}_{2j\rho}(P_1) \circ U_R(2j)),
\]

\[
R_2 := U(1) \circ \bigcirc_{j=1}^{\ell^2} (\operatorname{tr}_{2j\rho}(P_2) \circ U(2j + 1)).
\]

The figure to the right shows $R_1$ (dotted) and $R_2$ (solid) for $\ell^2 = 4$, see below for a figure showing $\ell^2 = 1$ with more details visible.

We denote by $R_{ij}$ the $j$-th “summand” of $R_1$, i.e., $R_{ij} = U_L(2j) \circ \operatorname{tr}_{2j\rho}(P_i) \circ U_R(2j)$. Informally, we will use the term $U$-shape for the subcurves $R_{ij}$ and $U(2j + 1)$, since they resemble the letter U. Moreover, we consider “summands” of $R_2$, namely $R_{ij} = U(2j - 1) \circ \operatorname{tr}_{2j\rho}(P_2) \circ ((2j + 1)\rho, 0)$ and $R_{ij} = ((2j - 1)\rho, 0) \circ \operatorname{tr}_{2j\rho}(P_2) \circ U(2j + 1)$.
Intuition  Considering traversals that stay within distance 1, we can traverse one \( U \)-shape in \( R_1 \) and one neighboring \( U \)-shape in \( R_2 \) together. Such traversals can be stitched together to a traversal of any number \( j \) of neighboring \( U \)-shapes in both curves. So far we can only traverse the same number of \( U \)-shapes in both curves, but \( R_2 \) has one more \( U \)-shape than \( R_1 \). We will show that we can traverse two \( U \)-shapes in \( R_2 \) while traversing only one \( U \)-shape in \( R_1 \), if these parts contain a satisfying assignment.

In the unsatisfiable case, essentially we show that we cannot traverse two \( U \)-shapes in \( R_2 \) while traversing only one \( U \)-shape in \( R_1 \), which implies a contradiction since the number of \( U \)-shapes in \( R_2 \) is larger than in \( R_1 \). We make this intuition formal in the remainder of this section.

Analysis  In order to be able to replace the curves \( P_1^j, P_2^j \) constructed above by other curves in the next section, we analyse the OR-gadget in a rather general way. To this end, we first specify a set of properties and show that the curves \( P_1^j, P_2^j \) constructed above satisfy these properties. Then we analyse the OR-gadget using only these properties of \( P_1^j, P_2^j \).

Property 4.1.  
(i) If \( \varphi \) is satisfiable then for some \( j \in [\ell^2] \) we have \( d_{df}(P_1^j, P_2^j) \leq 1 \).
(ii) If \( \varphi \) is not satisfiable then for all \( j \in [\ell^2] \) and curves \( \sigma_1, \sigma_2, \pi_1, \pi_2 \) such that \( \sigma_1 \) stays to the left and above \( (-\rho, \rho) \) and \( \pi_1 \) stays to the right and above \( (\rho, \rho) \), we have \( d_F(\sigma_1 \circ P_1^j \circ \pi_1, \sigma_2 \circ P_2^j \circ \pi_2) > \beta \), for some \( \beta > 1 \).
(iii) \( P_1^j \) is \( \Theta(c) \)-packed for some \( c \geq 1 \) for all \( j \in [\ell^2] \), \( k \in \{1, 2\} \).
(iv) \((0, \rho)\) is within distance 1 of any point in \( P_1^j \) for all \( j \in [\ell^2] \).
(v) \((0, 0)\) is within distance 1 of any point in \( P_2^j \) for all \( j \in [\ell^2] \).

Lemma 4.2.  
The curves \((P_1^j, P_2^j)\) constructed above satisfy Property 4.1 with \( \beta = 1.001 \) and \( c = \Theta(M \cdot 2^{N/2} / \ell) \). Moreover, we have \( |P_k^j| = \Theta(M \cdot 2^{N/2} / \ell) \) for all \( j \in [\ell^2] \), \( k \in \{1, 2\} \).

Proof. Property 4.1.(i) follows from Lemma 3.5, since at least one pair \((A_1^j, A_2^j)\) contains a satisfying assignment. Properties (iv) and (v) can be verified by considering all points in the construction in Section 3.1.

Observe that \( |P_k^j| = \Theta(M \cdot 2^{N/2} / \ell) \), since \( P_k^j \) consists of \( |A_k^j| = \Theta(2^{N/2} / \ell) \) assignment gadgets of size \( \Theta(M) \). The upper bound of (iii) follows since any polygonal curve with at most \( m \) segments is \( m \)-packed. The lower bound of (iii) follows from \( P_k^j \) being contained in a ball of radius 1 (by (iv) and (v)) and every segment of \( P_k^j \) having constant length.

For (ii), note that from any traversal of \((\sigma_1 \circ P_1^j \circ \pi_1, \sigma_2 \circ P_2^j \circ \pi_2)\) with width 1.001 one can extract a traversal of \((P_1^j, P_2^j)\) with width 1.001, by mapping any point in \( \sigma_k \) to the starting point \( s_k \) of \( P_k^j \) and any point in \( \pi_k \) to the endpoint \( t_k \) of \( P_k^j \), \( k \in \{1, 2\} \). This does not increase the width, since (1) \( s_2 \) and \( t_2 \) are within distance 1 to all points in \( P_1^j \), and (2) \( s_1 \) has smaller distance to any point in \( P_2^j \) than any point in \( \sigma_1 \) has, since \( \sigma_1 \) stays above and to the left of \( s_1 \), while all points of \( P_1^j \) lie below and to the right of \( s_1 \). A similar statement holds for \( t_1 \) and \( \pi_1 \). Property (ii) now follows from Lemma 3.7.  

In the following lemma we analyse the OR-gadget.
Lemma 4.3. For any curves \( (P_1^j, P_2^j) \) that satisfy Property 4.1, the OR-gadget \((R_1, R_2)\) satisfies:

(i) \(|R_k| = \Theta(\sum_{j=1}^{\ell^2} |P_k^j|) \) for \( k \in \{1, 2\} \).

(ii) \( R_1 \) and \( R_2 \) are \( \Theta(c) \)-packed.

(iii) If \( \varphi \) is satisfiable then \( d_F(R_1, R_2) \leq d_{df}(R_1, R_2) \leq 1 \).

(iv) If \( \varphi \) is not satisfiable then \( d_{df}(R_1, R_2) \geq d_F(R_1, R_2) > \min\{\beta, 1.2\} \).

Proof. (i) Precisely, we have \(|R_k| = \sum_{j=1}^{\ell^2} (|P_k^j| + 10) + 10(k - 1) \) for \( k \in \{1, 2\} \).

(ii) Let \( k \in \{1, 2\} \) and consider any ball \( B = B(q, r) \). If \( r \leq 1 \) then \( B \) hits \( O(1) \) of the curves \( P_k^j \). Since these curves are \( c \)-packed, their contribution to the total length of \( R_k \) in \( B \) is at most \( O(cr) \). Moreover, \( B \) hits \( O(1) \) segments of \( U \) or \( U_L, U_R \), and the connecting segments to \( P_k^j \). Each of these segments has length at most \( 2r \) inside \( B \). This yields a total length of \( R_k \) in \( B \) of \( O((c + 1)r) \).

Similarly, if \( r > 1 \) then \( B \) hits \( O(r) \) of the curves \( P_k^j \). Note that the total length of \( P_k^j \) is at most \( c \), since the curve is \( c \)-packed and contained in a ball of radius 1 around \((0, 0)\) or \((0, \rho)\) by Property 4.1. Hence, the total length of of the curves \( P_k^j \) in \( B \) is \( O(\alpha c) \). Moreover, \( B \) hits \( O(r) \) segments of \( U, U_L, U_R \), and the connectors to \( P_k^j \), each of constant length. This yields a total length of \( R_k \) in \( B \) of \( O((c + 1)r) \).

In total, the curve \( R_k \) is \( O(c + 1) \)-packed. As \( c \geq 1 \), it is also \( O(c) \)-packed. Since for some \( \alpha > 0 \) the curve \( P_k^j \) is not \( \alpha c \)-packed, also \( R_k \) is not \( \alpha c \)-packed, so \( R_k \) is even \( \Theta(c) \)-packed.

(iii) Note that \( d_F(R_1, R_2) \leq d_{df}(R_1, R_2) \) holds in general, so we only have to show that if \( \varphi \) is satisfiable then \( d_{df}(R_1, R_2) \leq 1 \). First we show that we can traverse one \( U \)-shape in \( R_1 \) and one neighboring \( U \)-shape in \( R_2 \) together.

Claim 4.4. For any \( j \in [\ell^2] \), we have \( d_{df}(R_1, U(2j - 1)) \leq 1 \) and \( d_{df}(R_1, U(2j + 1)) \leq 1 \).

Proof. We only show the first inequality, the second is similar. We start by traversing \( U_L(2j) \) and the left half of \( U(2j - 1) \) in parallel, being at the \( i \)-th point of \( U_L(2j) \) and \( U(2j - 1) \) at the same time. At any point in time we are within distance \( \rho \). Now we step to \( (2j + \rho, \rho) \) in \( U(2j - 1) \). We stay there while traversing \( tr_{2j\rho}(P_1^j) \) in \( P_1^j \), staying within distance 1 by Property 4.1.(iv). Finally, we traverse \( U_R(2j) \) and the second half of \( U(2j - 1) \) in parallel, where again the largest encountered distance is \( \rho \).

We can stitch these traversals together so that we traverse any number \( j \) of neighboring \( U \)-shapes in both curves together, because the parts in between the \( U \)-shapes are near to a single point, as shown by the following claim. Note that \( (2j + \rho, 0) \circ ((2j + 2)\rho, 0) \) is the connecting segment in \( R_1 \) between \( U_R(2j) \) and \( U_L(2j + 2) \), while \( ((2j - 1)\rho, 0) \circ tr_{2j\rho}(P_1^j) \circ ((2j + 1)\rho, 0) \) is the part in \( R_2 \) between \( U(2j - 1) \) and \( U(2j + 1) \).

Claim 4.5. For any \( j \in [\ell^2] \),

\[
\begin{align*}
d_{df}((2j\rho, 0) \circ ((2j + 2)\rho, 0), ((2j + 1)\rho, 0)) &\leq 1, \\
d_{df}((2j\rho, 0), ((2j - 1)\rho, 0) \circ tr_{2j\rho}(P_1^j) \circ ((2j + 1)\rho, 0)) &\leq 1.
\end{align*}
\]

Proof. The first claim is immediate. The second follows from Property 4.1.(v).

Thus, we can stitch together traversals of \( U \)-shapes in both curves. However, so far we can only traverse the same number of \( U \)-shapes in both curves, but \( R_2 \) has one more \( U \)-shape than \( R_1 \).
Consider $J \in [\ell^2]$ with $d_{\text{AF}}(P^I_1, P^I_2) \leq 1$, which exists since $\varphi$ is satisfiable, see Property 4.1.(i).

Consider the two subcurves (also see the above figure)

$$
R'_1 := R'_1 = U_L(2J) \circ \text{tr}_{2J\rho}(P^I_1) \circ U_R(2J),
$$
$$
R'_2 := U(2J - 1) \circ \text{tr}_{2J\rho}(P^I_2) \circ U(2J + 1).
$$

We show that $d_{\text{AF}}(R'_1, R'_2) \leq 1$, i.e., we can traverse two $U$-shapes in $R_2$ while traversing only one $U$-shape in $R_1$, using $d_{\text{AF}}(P^I_1, P^I_2) \leq 1$. Adding simple traversals of $U$-shapes before and after $(R'_1, R'_2)$, we obtain a traversal of $(R_1, R_2)$ with width 1, proving $d_{\text{AF}}(R_1, R_2) \leq 1$. It is left to show the following claim.

**Claim 4.6.** $d_{\text{AF}}(R'_1, R'_2) \leq 1$.

**Proof.** We traverse $U_L(2J)$ and $U(2J - 1)$ in parallel until we are at point $((2J - 1)\rho, 2\rho)$ in $U_L(2J)$. We stay in this point and follow $U(2J - 1)$ until its second-to-last point. In the next step we can finish traversing $U_L(2J)$ and $U(2J - 1)$. In the next step we go to the first positions of (the translated) $P^I_1$ and $P^I_2$. We follow any traversal of $(P^I_1, P^I_2)$ with width 1. Finally, we use a traversal symmetric to the one of $(U_L(2J), U(2J - 1))$ to traverse $(U_R(2J), U(2J + 1))$. 

(iv) Note that the inequality $d_{\text{AF}}(R_1, R_2) \geq d_{\text{F}}(R_1, R_2)$ holds in general, so we only have to show that if $\varphi$ is not satisfiable then $d_{\text{F}}(R_1, R_2) > \min\{\beta, 1.2\}$. Assume for the sake of contradiction that there is a traversal of $(R_1, R_2)$ with width $\min\{\beta, 1.2\}$. Essentially we show that it cannot traverse 2 $U$-shapes in $R_2$ while traversing only one $U$-shape in $R_1$, which implies a contradiction since the number of $U$-shapes in $R_2$ is larger than in $R_1$.

Let $Y_\rho$ be the line $\{(x, y) \in \mathbb{R}^2 | y = \rho\}$. We inductively prove the following claims.

**Claim 4.7.**

(i) For any $0 \leq j \leq \ell^2$, when the traversal is in $R_2$ at the left highest point $(2j\rho, 3\rho)$ of $U(2j + 1)$, then in $R_1$ we fully traversed $R^I_1$ and are above the line $Y_\rho$.

(ii) For any $1 \leq j \leq \ell^2$, when the traversal is in $R_1$ at the right highest point $((2j + 1)\rho, 3\rho)$ of $R^I_1$, then in $R_2$ it is in $U(2j - 1)$.

Note that claim (i) for $j = \ell^2$ yields the desired contradiction, since after traversing $R^{\ell^2}_1$ the curve $R_1$ has ended (at the point $(2\ell^2\rho, 0)$), so that we cannot go above the line $Y_\rho$ anymore.

**Proof.** (i) Note that we have to be above the line $Y_\rho$ because all points below $Y_\rho$ have distance at least $2\rho > 1.2$ to the point $(2j\rho, 3\rho)$. For $j = 0$, claim (i) holds immediately, since there is no subcurve $R^I_0$ (so this part of the statement disappears). In general, claim (i) for any $1 \leq j \leq \ell^2$ follows from claim (ii) for $j$: When we are at $z_1 := ((2j + 1)\rho, 3\rho)$ in $R^I_1$, we are still in $U(2j - 1)$. Once we reach the endpoint $z_2 := ((2j - 1)\rho, 0)$ of $U(2j - 1)$, in $R_1$ we are at a point $p_1$ below the line $Y_\rho$, since all points in $R_2$ that follow $z_1$ and lie above $Y_\rho$ have distance more than $2\rho > 1.2$ to $z_2$. Now we follow $R_2$ until we reach $p_2 := (2j\rho, 3\rho)$ in $U(2j + 1)$. At this point we have to be above the line $Y_\rho$ in $R_1$, but all points in $R^I_1$ following $p_1$ lie below $Y_\rho$. Thus, at this point we have fully traversed $R^I_1$ (and have to be in $R^{I+1}_1$).

(ii) This claim for any $1 \leq j \leq \ell^2$ follows from claim (i) for $j - 1$. Assume for the sake of contradiction that claim (ii) for some $j$ does not hold. Consider the subcurve $R'_1$ of $R^I_1$ between (the first occurrence of) $((2j - 1)\rho, \rho)$ and $((2j + 1)\rho, 3\rho)$. Let $R'_2$ be the subcurve of $R_2$ that the traversal traverses together with $R'_1$. Since $(R'_1, R'_2)$ forms a subtraversal of the traversal of $(R_1, R_2)$, which has width $\min\{\beta, 1.2\}$, we have $d_{\text{F}}(R'_1, R'_2) \leq \min\{\beta, 1.2\}$ (*). By claim (i) for $j - 1$, the starting point of $R'_2$ lies before $\text{tr}_{2j\rho}(P^I_2)$ along $R_2$, since we reach $((2j - 2)\rho, 3\rho)$ in
Thus, any $U(2j - 1)$ only after being in the starting point of $R_1^j$. Moreover, the endpoint of $R_2^j$ lies after $\text{tr}_{2j\rho}(P_2^j)$ along $R_2$. Indeed, while being at the endpoint $((2j + 1)\rho, 3\rho)$ of $R_1^j$, we cannot be in $U(2j - 1)$ since we assumed that claim (ii) is wrong for $j$. We can also not be in $\text{tr}_{2j\rho}(P_2^j)$, since by Property 4.1.5 all points in this curve lie in a ball of radius 1 around $((2j + 1)\rho, 3\rho)$, so their distance to $((2j + 1)\rho, 3\rho)$ is at least $\|(2j + 1)\rho, 3\rho) - (2j\rho, 0)\| = \sqrt{5} - 1 > 1.2$. Hence, we already passed $\text{tr}_{2j\rho}(P_2^j)$, and $R_2^j$ is of the form $\sigma_2 \circ \text{tr}_{2j\rho}(P_2^j) \circ \pi_2$ for any curves $\sigma_2, \pi_2$. Note that $R_1^j$ is of the form $\sigma_1 \circ \text{tr}_{2j\rho}(P_1^j) \circ \pi_1$ with $\sigma_1$ staying above and to the left of $((2j - 1)\rho, \rho)$ and $\pi_1$ staying above and to the right of $((2j + 1)\rho, \rho)$. Thus, after translation Property 4.1.(ii) applies, proving $d_F(R_1^j, R_2^j) > \beta$, a contradiction to (*).

\[
\gamma - \frac{\delta}{2} \leq c \leq (n^\gamma + \delta^2)^{N/2}.
\]

From this it follows that $\Omega(n^{\gamma - \delta/2}) \leq c \leq O(n^{\gamma + \delta/2})$, which implies the desired polynomial restriction $n^{\gamma - \delta} \leq c \leq n^{\gamma + \delta}$ for sufficiently large $n$.

4.2 Approximation schemes

In this section, we consider the dependence on $\varepsilon$ of the runtime of a $(1 + \varepsilon)$-approximation for the Fréchet distance on $c$-packed curves. We show that in $\mathbb{R}^d$ with $d \geq 5$ there is no such algorithm with runtime $O(\min\{cn/\sqrt{\varepsilon}, n^2\}^{1-\delta})$ for any $\delta > 0$ unless SETH fails (Theorem 1.4). This matches the dependence on $\varepsilon$ of the fastest known algorithm up to a polynomial. The result holds for sufficiently small $\varepsilon > 0$ and any polynomial restriction of $1 \leq c \leq n$ and $\varepsilon \leq 1$.

We will reuse the OR-gadget from the last section, embedded into the first two dimensions of $\mathbb{R}^5$. Specifically, we will reuse Lemma 4.3. However, we adapt the curves $P_1^j, P_2^j$, essentially by embedding the same set of points in a different way.

In this new embedding we make use of the fact that in $\mathbb{R}^4$ there are point sets $Z_1, Z_2$ of arbitrary size such that any pair of points $(z_1, z_2) \in Z_1 \times Z_2$ has distance 1. For an example, see the figure to the right, where the left picture shows the projection onto the first two dimensions and the right picture shows the projection onto the last two dimensions. Here, $Z_1$ (circles) is placed along a quarter-circle in the $(1, 2)$-plane and $Z_2$ (crosses) is placed along a quarter-circle in the $(3, 4)$-plane.
Construction  As usual, consider a CNF-SAT instance $\varphi$, partition its variables $x_1, \ldots, x_N$ into two sets $V_1, V_2$ of size $N/2$, and consider any set $A_k$ of assignments of $T$ and $F$ to the variables in $V_k$. Fix any enumeration $\{a_1^k, \ldots, a_{|A_k|}^k\}$ of $A_k$. Again, we split the considered points into two sets, as in the above figure. In particular, $A_k$ corresponds to the clause gadgets of $A_k$.

Thus, we align the clause gadgets of $A_k$. We define assignment gadgets and the curves $P, Q, R$ as in Section 3.1, i.e.,

$$\begin{align*}
AG(a_k) & := r_k(a_k) \circ \bigcirc_{i \in [M]} CG(a_k, i), \\
P_1 & := \bigcirc_{a_1 \in A_1} \left( s_1(a_1) \circ AG(a_1) \circ t_1(a_1) \right), \\
P_2 & := s_2 \circ s_* \circ \left( \bigcirc_{a_2 \in A_2} AG(a_2) \right) \circ t_* \circ t_2.
\end{align*}$$

Analysis  Again, we split the considered points into $Q_1, Q_2$, depending on whether they may appear on $P_1$ or $P_2$, i.e., $Q_1 := \{s_1(a_1), t_1(a_1), r_1(a_1), CG(a_1, i) \mid a_1 \in A_1, i \in [M]\}$ and $Q_2 := \{s_2, t_2, s_*^k, t_*^k, r_2(a_2), CG(a_2, i) \mid a_2 \in A_2, i \in [M]\}$. It is easy, but tedious to verify that the constructed points behave as follows.

Lemma 4.8. The following pairs of points have distance at most 1 for any $a_k \in A_k$:

$$(q, s_2), (q, t_2) \text{ for any } q \in Q_1,$$
$$(s_1(a_1), q) \text{ for any } q \in Q_2 \setminus \{t_*^k\},$$
$$(t_1(a_1), q) \text{ for any } q \in Q_2 \setminus \{s_*^k\},$$
$$(r_1(a_1), r_2(a_2)),$$
$$(CG(a_1, i), CG(a_2, i)) \text{ if assignment } (a_1, a_2) \text{ satisfies clause } C_i.$$
Moreover, the following pairs of points have distance more than \(1 + \varepsilon\) for any \(a_k \in A_k\):

\[
\begin{align*}
(q, s^*_2) & \text{ for any } q \in Q_1 \setminus \{s_1\}, \\
(q, t^*_2) & \text{ for any } q \in Q_1 \setminus \{t_1\}, \\
(r_1(a_1), CG(a_2, i)) & \text{ for any } i \in [M], \\
(CG(a_1, i), r_2(a_2)) & \text{ for any } i \in [M], \\
(CG(a_1, i), CG(a_2, j)) & \text{ for any } i, j \in [M], i \not\equiv j \mod 2, \\
(CG(a_1, i), CG(a_2, i)) & \text{ if assignment } (a_1, a_2) \text{ does not satisfy clause } C_i.
\end{align*}
\]

**Proof.** Using that \(\varepsilon\) is sufficiently small, we only have to compute the largest order term of \(\varepsilon\) for all distances. E.g., for all \(a_k \in A_k\)

\[
\|s_1(a_1) - r_2(a_2)\| = \sqrt{\rho^2((1 - 400\varepsilon)^2 + 1 + (18\sqrt{\varepsilon})^2)} = \sqrt{1 - 476\varepsilon + \mathcal{O}(\varepsilon^2)} \leq 1.
\]

Now we use these curves in the OR-gadget from the last section. To this end, again partition the set of all assignments of \(V_k\) into sets \(A_k^1, \ldots, A_k^N\) of size \(\Theta(2^{N/2}/\ell)\), where we fix \(1 \leq \ell \leq 2^{N/2}\) later. Use the above construction of \(P_1, P_2\) after replacing \(A_1\) by \(A_1^1\) and \(A_2\) by \(A_2^1\) for any \(j_1, j_2 \in [\ell]\) to obtain curves \(P_{j_1j_2}^1, P_{j_1j_2}^2\). Slightly rename these curves so that we have curves \((P_{j_1j_2}^1, P_{j_1j_2}^2)\) for \(j \in \ell^2\). Then these curves satisfy Property 4.1.

**Lemma 4.9.** The curves \(P_{j_1}^1, P_{j_2}^2\) satisfy Property 4.1 with \(c = \Theta(1 + \sqrt{\varepsilon}M|A_k|)\) and \(\beta = 1 + \varepsilon\). Moreover, \(|P_{k}^j| = \Theta(M2^{N/2}/\ell)\) for any \(j \in \ell^2\), \(k \in \{1, 2\}\).

**Proof.** Using Lemma 4.8, we can follow the proof in Section 3.1, since everything that we used about \(P_1, P_2\) is captured by this lemma. This proves that if \(\varphi\) is satisfiable then \(\mathsf{d}_{\mathsf{DF}}(P_{j_1}^1, P_{j_2}^2) \leq 1\) for some \(j \in \ell^2\), and if \(\varphi\) is not satisfiable then \(\mathsf{d}_{\mathsf{DF}}(P_{j_1}^1, P_{j_2}^2) > 1 + \varepsilon\) for all \(j \in \ell^2\), i.e., Property 4.1.(i) and (ii) in the discrete case. The same adaptations as in Section 3.2 allow to prove correctness in the continuous case, we omit the details.

It is easy to see that all constructed points lie within distance 1 of \((0, 0, 0, 0, 0)\), showing (iv). For (v) we use that we placed the points along the upper quarter-circle, and not the full circle. This way, all points in \(P_{j}^1\) have a distance to \((0, \rho, 0, 0, 0)\) of at most \(\|(0, \rho) - (\frac{\rho}{\sqrt{2}}, 0)\| + \mathcal{O}(\sqrt{\varepsilon}) < 1\), for sufficiently small \(\varepsilon\).

For (iii) observe that all segments of \(P_{j}^1\) (except for the finitely many segments incident to \(s_2^*, t_2^*\)) have length \(\Theta(\sqrt{\varepsilon} + 1/(M|A_k^j|))\), \(k \in \{1, 2\}\). Moreover, the \(\Theta(M|A_k^j|)\) segments of \(P_{j}^1\) are spread along a quarter-circle. Hence, any ball \(B(q, r)\) intersects \(\mathcal{O}(1 + \min\{1, r\}M|A_k^j|)\) segments of \(P_{j}^1\). Since each of these segments has length \(\mathcal{O}(\min\{r, \sqrt{\varepsilon} + 1/(M|A_k^j|)\})\) in \(B(q, r)\), the total length of \(P_{j}^1\) in \(B(q, r)\) is \(\mathcal{O}(r(1 + \sqrt{\varepsilon}M|A_k^j|))\). Thus, \(P_{j}^1\) is \(\Theta(1 + \sqrt{\varepsilon}M|A_k^j|)\)-packed. It is also \(\Theta(1 + \sqrt{\varepsilon}M|A_k^j|)\)-packed, since all \(\Theta(M|A_k^j|)\) segments of length \(\Theta(\sqrt{\varepsilon} + 1/(M|A_k^j|))\) lie in a ball of radius 1 around \((0, 0, 0, 0, 0)\) or \((0, \rho, 0, 0, 0)\) by (iv) and (v). Finally, note that \(|A_k^j| = 2^{N/2}/\ell\). \(\blacksquare\)

**Proof of Theorem 1.4** The above Lemma 4.9 allows to apply Lemma 4.3, which constructs curves \(R_1, R_2\) such that any \((1 + \varepsilon)\)-approximation for the Fréchet distance of \((R_1, R_2)\) decides satisfiability of \(\varphi\). Since \(R_1\) and \(R_2\) are \(c\)-packed with

\[
c = \Theta(1 + \sqrt{\varepsilon}M2^{N/2}/\ell), \quad n = \max\{|R_1|, |R_2|\} = \Theta(\ell M 2^{N/2}),
\]

18
we obtain that any $(1 + \varepsilon)$-approximation for the Fréchet distance with runtime $O((cn/\sqrt{\varepsilon})^{1-\delta})$ yields an algorithm for CNF-SAT with runtime $O(M^22^{(1-\delta)N})$, as long as $\ell = O(\sqrt{\varepsilon}M2^{N/2})$. This contradicts SETH'.

Moreover, using Lemma 2.1 we can assume that $1 \leq M \leq 2^{4N/4}$. Setting $\ell := \Theta(\varepsilon n^{1/2(1+\gamma)/2}2^{(1+\gamma)/2}N/2)$ for any $0 \leq \gamma \leq 1$, we obtain

$$
\varepsilon n^{1/2(1+\gamma)/2}2^{(1+\gamma)/2}N/2 \leq c \leq \varepsilon n^{1/2(1+\gamma)/2}2^{(1+\gamma)/2}N/2.
$$

From this it follows that $\Omega(n^{\gamma-\delta/2}) \leq c \leq O(n^{\gamma+\delta})$, which implies the desired polynomial restriction $n^{\gamma-\delta} \leq c \leq n^{\gamma+\delta}$ for sufficiently large $n$. Note that this works as long as

$$
1 \leq \ell \leq O(\sqrt{\varepsilon}M2^{N/2}).
$$

Since $\ell = \Theta((n/c)^{1/2})$, the first inequality is equivalent to $cn/\sqrt{\varepsilon} \leq n^2$, which is a natural condition, since otherwise the exact algorithm for general curves is faster. Plugging in the definition of $\ell = \Theta(\varepsilon n^{1/2(1+\gamma)/2}2^{(1+\gamma)/2}N/2)$, the second inequality becomes $1/\varepsilon \leq (2N M(1+\gamma)/\gamma)^2$. Since $(1+\gamma)/\gamma \geq 2$, $n = O(\ell M^{2N/2}) \leq O(M^22^{N})$, and $c \geq 1$, this is implied by the first condition $cn/\sqrt{\varepsilon} \leq n^2$. Hence, we may choose any sufficiently small $\varepsilon = \varepsilon(n)$ with $cn/\sqrt{\varepsilon} \leq n^2$.

## 5 Conclusion

We presented strong evidence that the (continuous or discrete) Fréchet distance has no strongly subquadratic algorithms, by relating this problem to the Strong Exponential Time Hypothesis.

Our extensions of this main result include approximation algorithms and realistic input curves ($c$-packed curves). These extensions leave three particularly interesting open questions, asking for new algorithms or improved lower bounds. Here, we use $\tilde{O}$ to ignore any polylogarithmic factors in $n$, $c$, and $1/\varepsilon$.

1. Is there a strongly subquadratic $O(1)$-approximation for the Fréchet distance on general curves?

2. In any dimension $d \in \{2, 3, 4\}$, is there a $(1 + \varepsilon)$-approximation with runtime $\tilde{O}(cn)$ for the Fréchet distance on $c$-packed curves? Or is there even an exact algorithm with runtime $\tilde{O}(cn)$?

3. In any dimension $d \geq 5$, is there a $(1 + \varepsilon)$-approximation with runtime $\tilde{O}(cn/\sqrt{\varepsilon})$ for the Fréchet distance on $c$-packed curves?

**Acknowledgements** The author wants to thank Wolfgang Mulzer for introducing him to the problem, Marvin Kühnemann for useful discussions, Nabil H. Mustafa for his enthusiasm and encouragement, and an anonymous referee for helpful comments and pointers to the orthogonal vectors problem.

**References**


