RUB

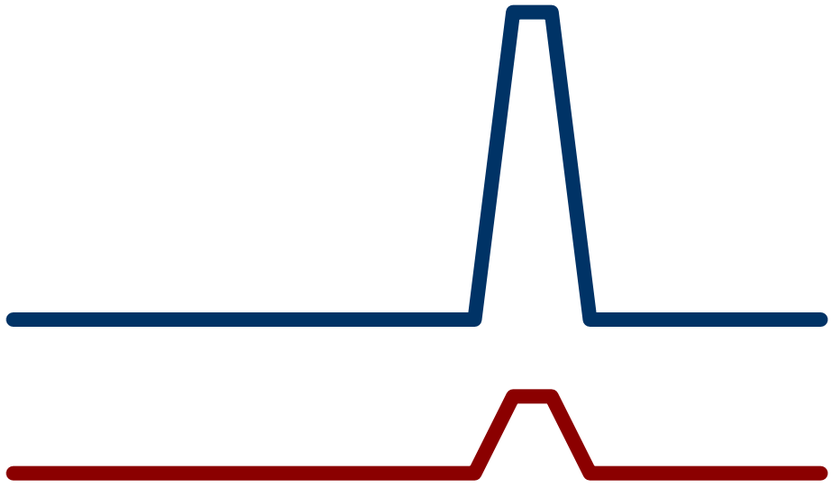
A natural metric for curves – 35 years of Fréchet distance computation

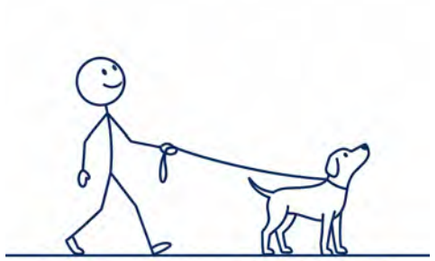
Maike Buchin

EuroCG 2026, Hagen

RUB

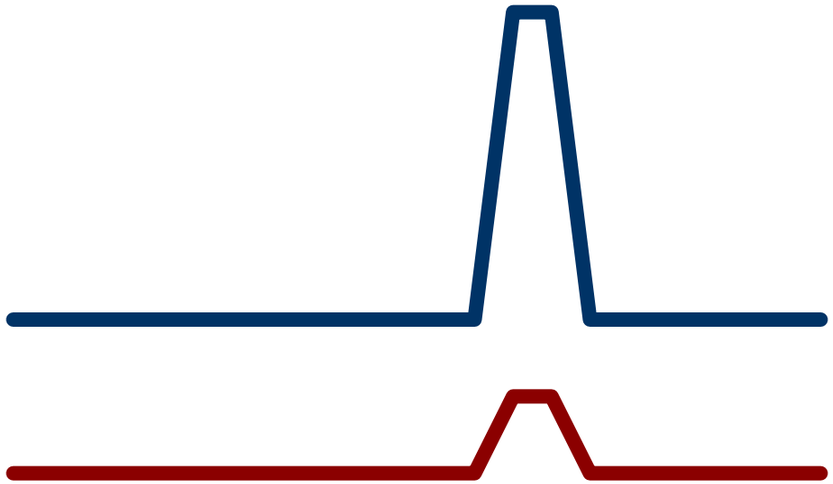
Question: How similar are two polygonal curves?

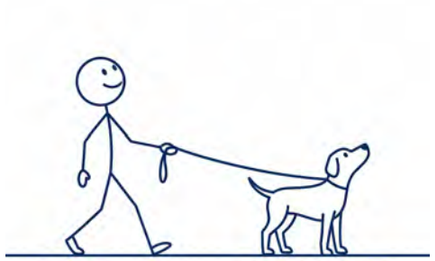


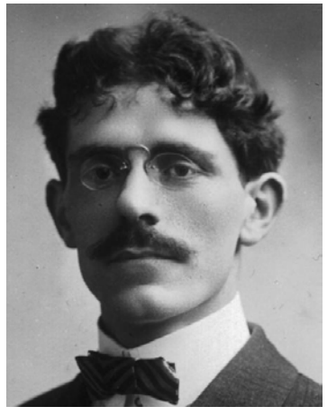


Introduction

An example of a metric for parameterized curves







Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

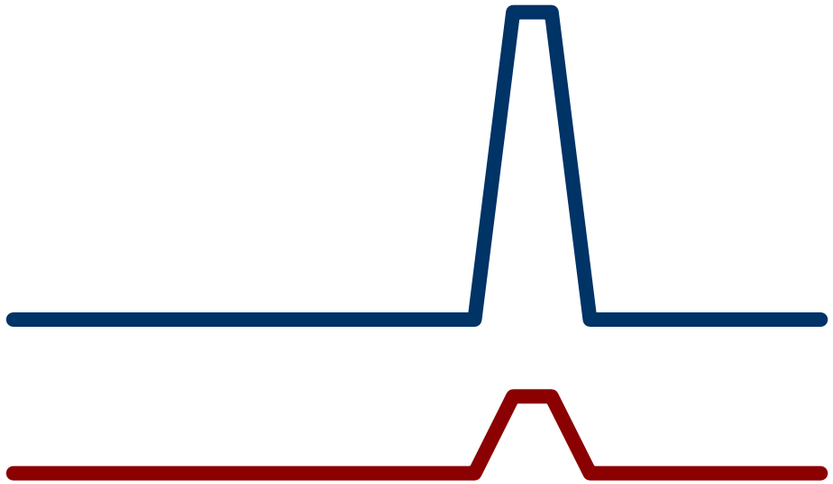
Definition:

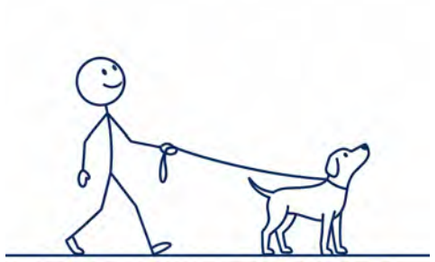
INF over *reparametrisation*, MAX over *distance betw. points*

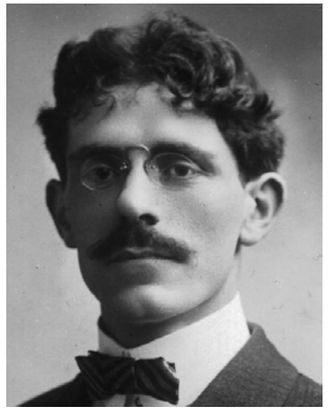
$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

homeomorphism







Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

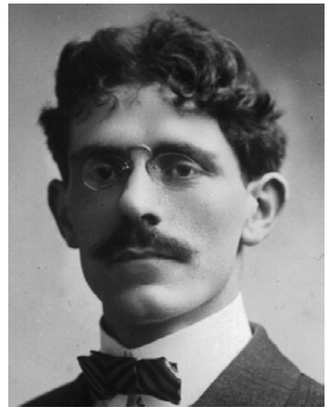
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

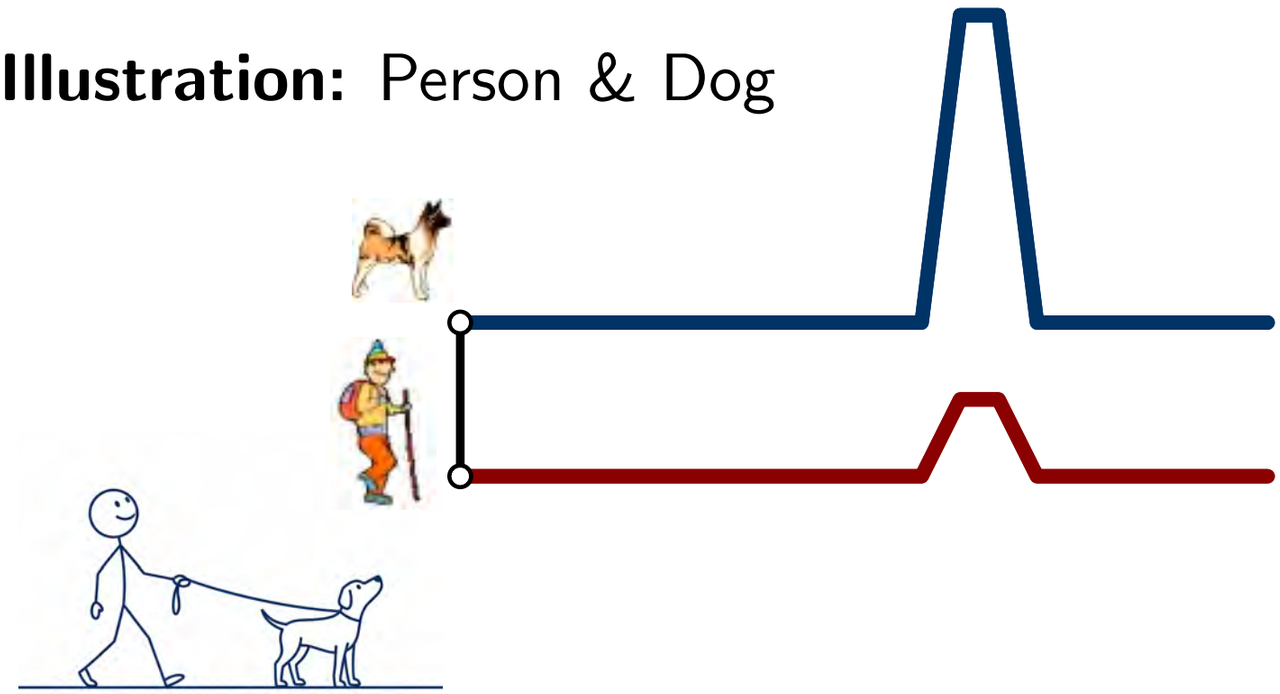
homeomorphism



Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog



SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

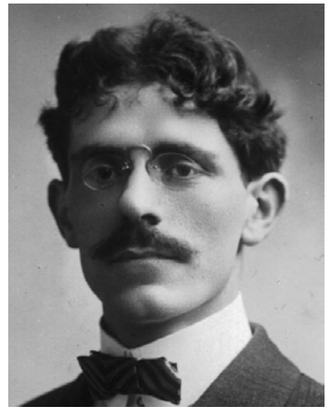
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

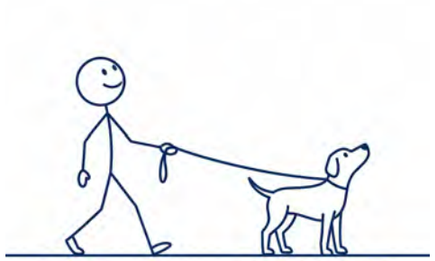
homeomorphism

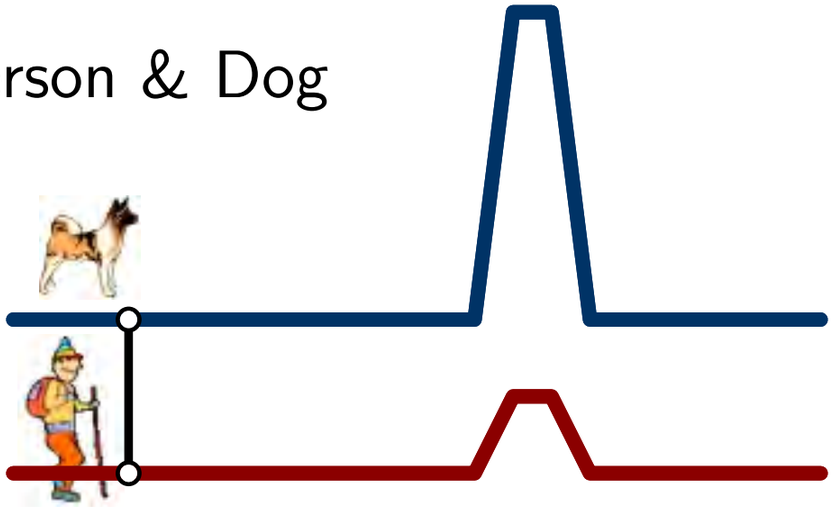


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog





SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

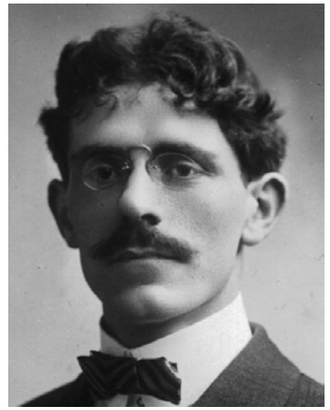
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

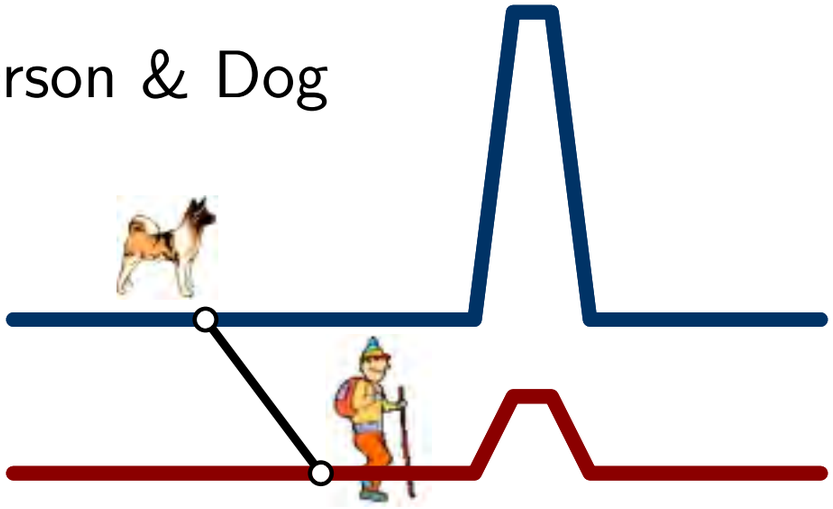
homeomorphism

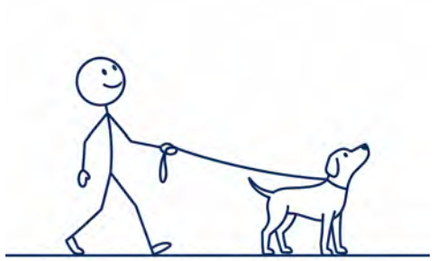


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog





SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

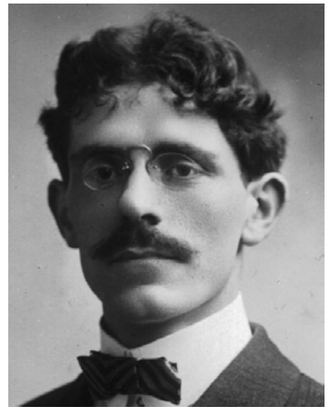
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

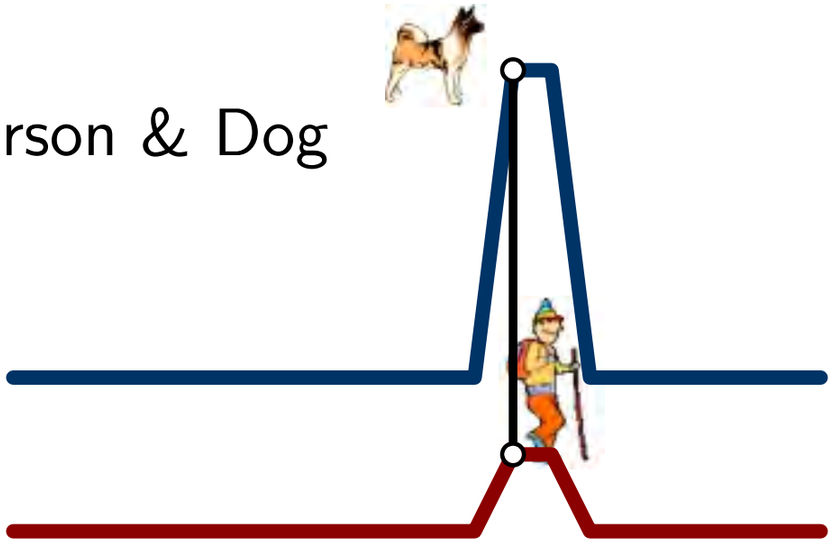
homeomorphism

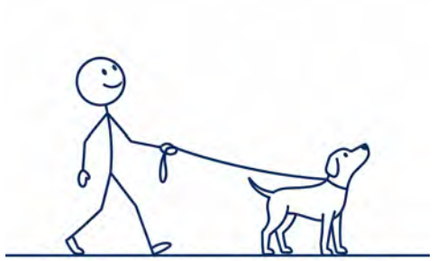


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog





SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

RUB

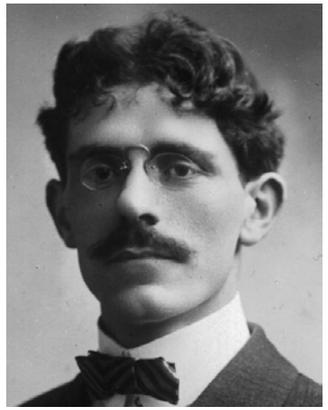
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

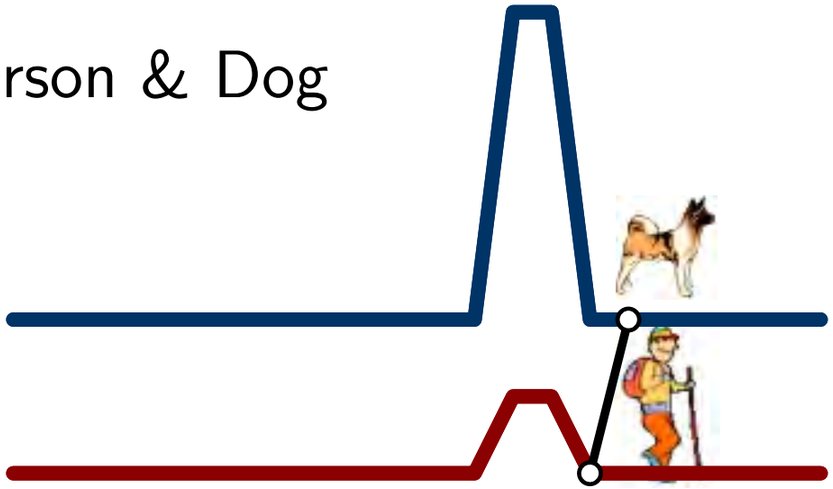
homeomorphism

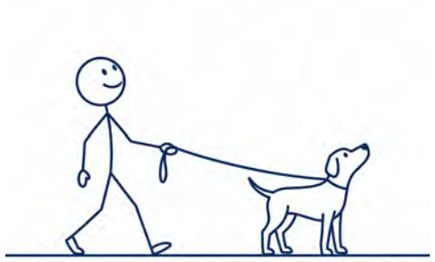


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog





SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

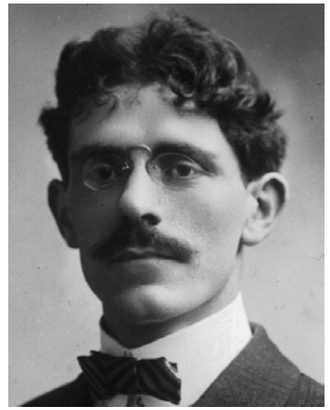
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

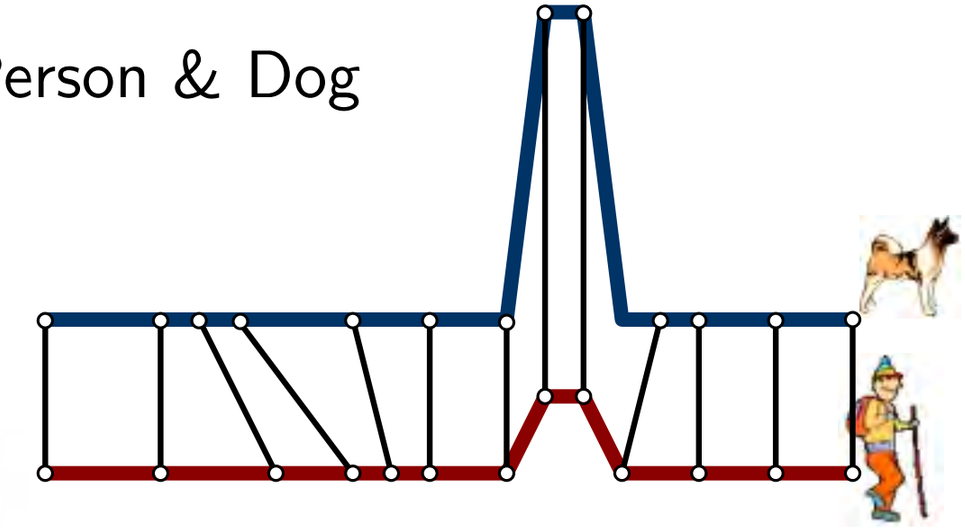
homeomorphism

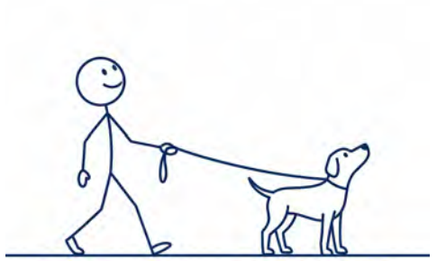


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog





SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

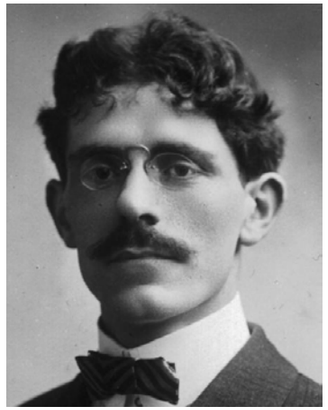
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

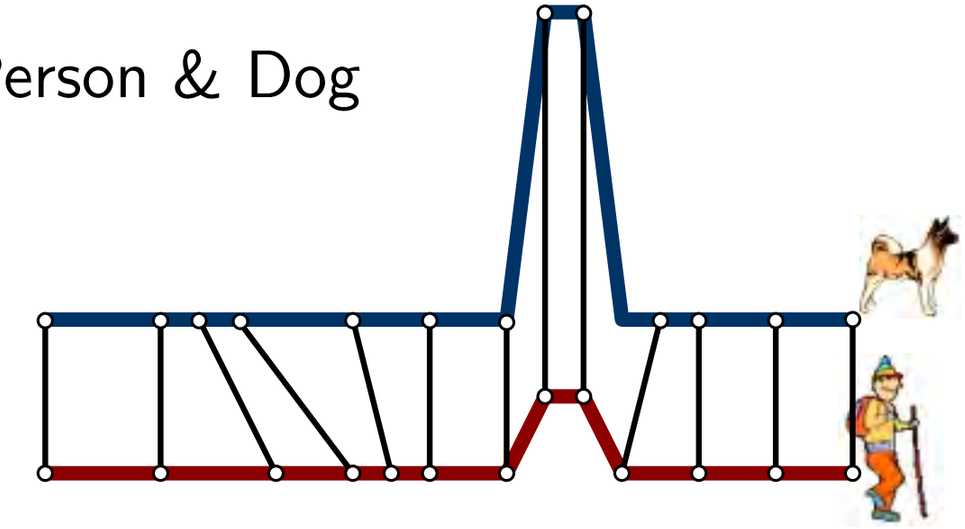
homeomorphism



Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

Illustration: Person & Dog



Fréchet distance equals shortest leash length

SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Introduction

RUB

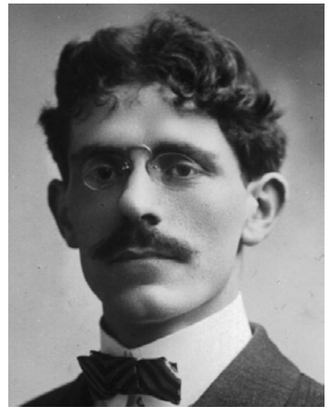
Definition:

INF over *reparametrisation*, MAX over *distance betw. points*

$P, Q\colon [0,n] \to \mathbb{R}^d$ parametrised curves

$$d_F(P,Q) := \inf_{\sigma:[0,n]\to[0,n]} \max_{x\in[0,n]} d(P(x), Q(\sigma(x)))$$

σ: homeomorphism

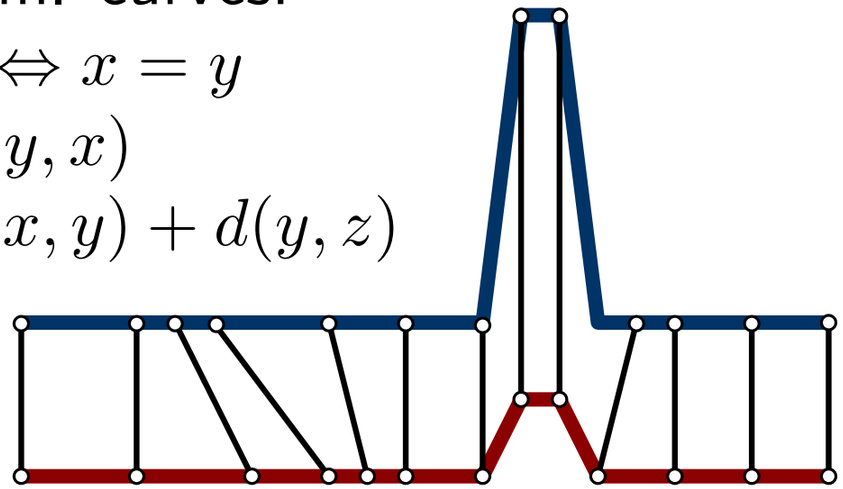


Maurice Fréchet, 1906

https://perso.lpsm.paris/ mazliak/ConfFrechetang.html

metric for param. curves:

- $d(x,y) = 0 \Leftrightarrow x = y$
- $d(x,y) = d(y,x)$
- $d(x,z) \leq d(x,y) + d(y,z)$



SUR QUELQUES POINTS DU CALCUL FONCTIONNEL;

Par M. **Maurice Fréchet** (Paris) *).

Adunanza del 22 aprile 1906.

INTRODUCTION.

1. On s'accorde assez généralement à considérer un nombre y comme *fonction* de la variable x, quand à toute valeur numérique de x correspond une valeur bien déterminée de y. Lorsqu'on impose à cette correspondance certaines conditions de continuité, dérivation, etc., on obtient des classes de fonctions jouissant de propriétés précises. C'est l'étude de ces propriétés qui constitue le principal objet de l'Analyse moderne.

Research on computing the Fréchet distance

RUB

in 1992 Helmut Alt and Michael Godau presented the first algorithm

A natural metric for curves — Computing the distance for polygonal chains and approximation algorithms

Michael Godau*
godau@leibniz.math.fu-berlin.de
Freie Universität Berlin, FB Mathematik, WE 03
Arnimallee 2–6, D-1000 Berlin 33

Abstract

The often explored problem to approximate a given po
considered from a computational geometric point of view or
model it reasonably we give a natural definition of the di
Furthermore we give algorithms to calculate this distance
chains in the d-dimensional space for arbitrary d. With kno
polynomial time algorithms to approximate polygonal chains
an optimal solution under some constraints. We will show t
by a constant factor worse than the global optimum.

COMPUTING THE FRÉCHET DISTANCE BETWEEN TWO POLYGONAL CURVES *

HELMUT ALT and MICHAEL GODAU
Freie Universität Berlin, Fachbereich Mathematik und Informatik, Takustraße 9
14195 Berlin, Germany
e-mail: alt@inf.fu-berlin.de

Measuring the Resemblance of Polygonal Curves *

(Extended Abstract)

Helmut Alt Michael Godau †

trary dimensions we consider
arametrizations of the curves.
lges an algorithm of runtime
' and Q is developed. Then
net-distance for closed curves,
on derived from the Fréchet-
curve Q.

1 Introduction, Definitions

In many applications, two-dimensional *shapes* are given by the planar curves forming their boundaries. Consequently, a natural problem in shape comparison and recognition is to measure, how much two given curves "resemble each other". Naturally, the first question to be answered, is what metric between curves should be used to reflect the intuitive notion of "resemblance".

Figure 1: Two curves with HAUSDORFF-distance δ.

Research on computing the Fréchet distance

RUB

in 1992 Helmut Alt and Michael Godau presented the first algorithm

A natural metric for curves — Computing the distance for polygonal chains and approximation algorithms

Michael Godau*
godau@leibniz.math.fu-berlin.de
Freie Universität Berlin, FB Mathematik, WE 03
Arnimallee 2–6, D-1000 Berlin 33

Abstract

The often explored problem to approximate a given po
considered from a computational geometric point of view or
model it reasonably we give a natural definition of the di
Furthermore we give algorithms to calculate this distance
chains in the d-dimensional space for arbitrary d. With kno
polynomial time algorithms to approximate polygonal chains
an optimal solution under some constraints. We will show t
by a constant factor worse than the global optimum.

COMPUTING THE FRÉCHET DISTANCE BETWEEN TWO POLYGONAL CURVES *

HELMUT ALT and MICHAEL GODAU
Freie Universität Berlin, Fachbereich Mathematik und Informatik, Takustraße 9
14195 Berlin, Germany
e-mail: alt@inf.fu-berlin.de

Measuring the Resemblance of Polygonal Curves *

(Extended Abstract)

Helmut Alt Michael Godau †

trary dimensions we consider
arametrizations of the curves.
lges an algorithm of runtime
' and Q is developed. Then
net-distance for closed curves,
on derived from the Fréchet-
curve Q.

1 Introduction, Definitions

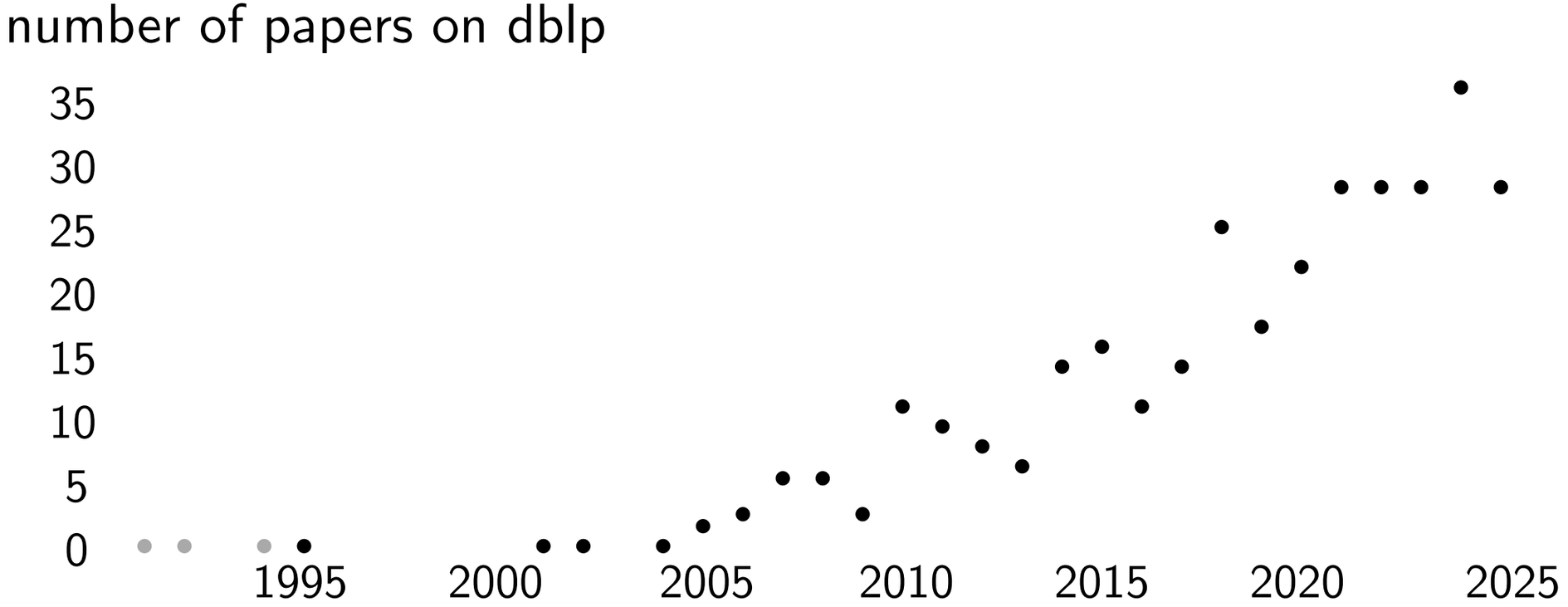
In many applications, two-dimensional *shapes* are given by the planar curves forming their boundaries. Consequently, a natural problem in shape comparison and recognition is to measure, how much two given curves "resemble each other". Naturally, the first question to be answered, is what metric between curves should be used to reflect the intuitive notion of "resemblance".

Figure 1: Two curves with HAUSDORFF-distance δ.

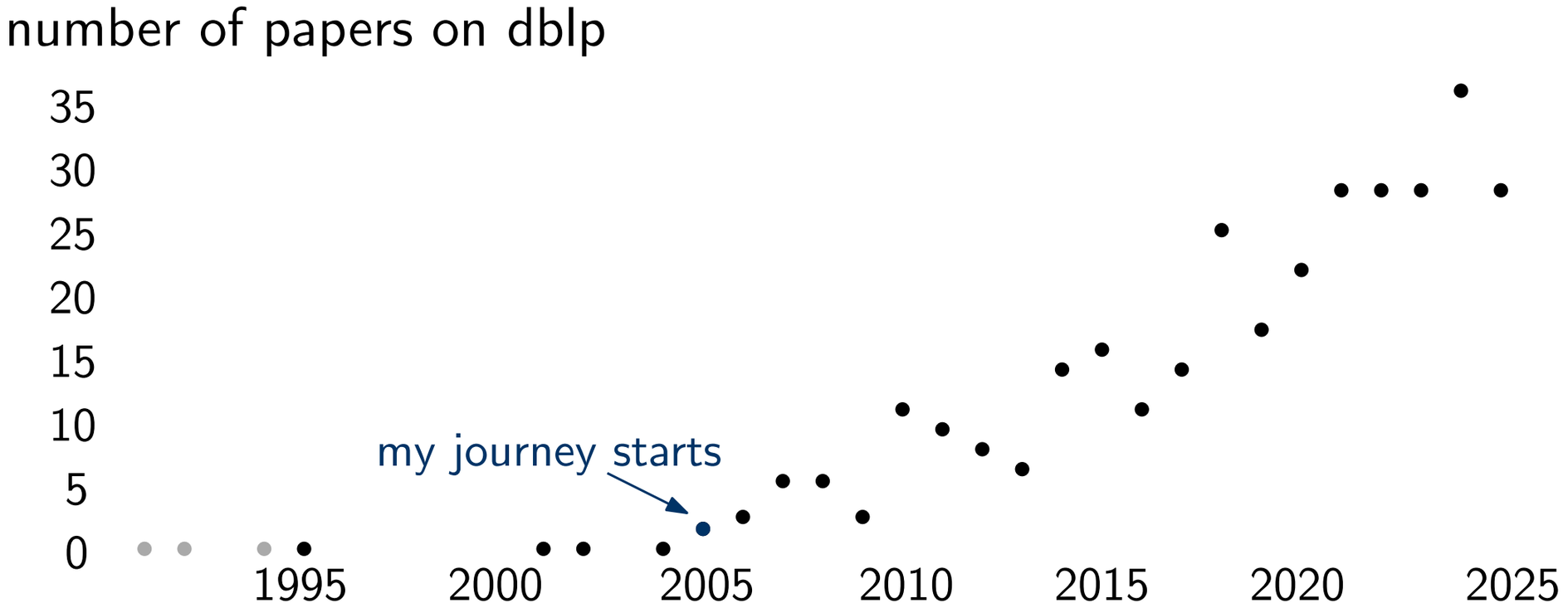
and many more papers followed..

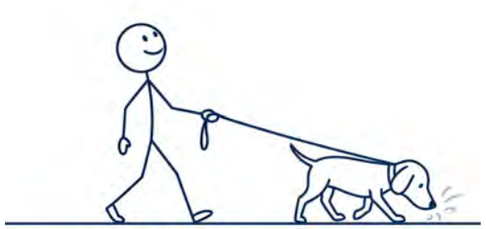
Research on computing the Fréchet distance

RUB



Research on computing the Fréchet distance

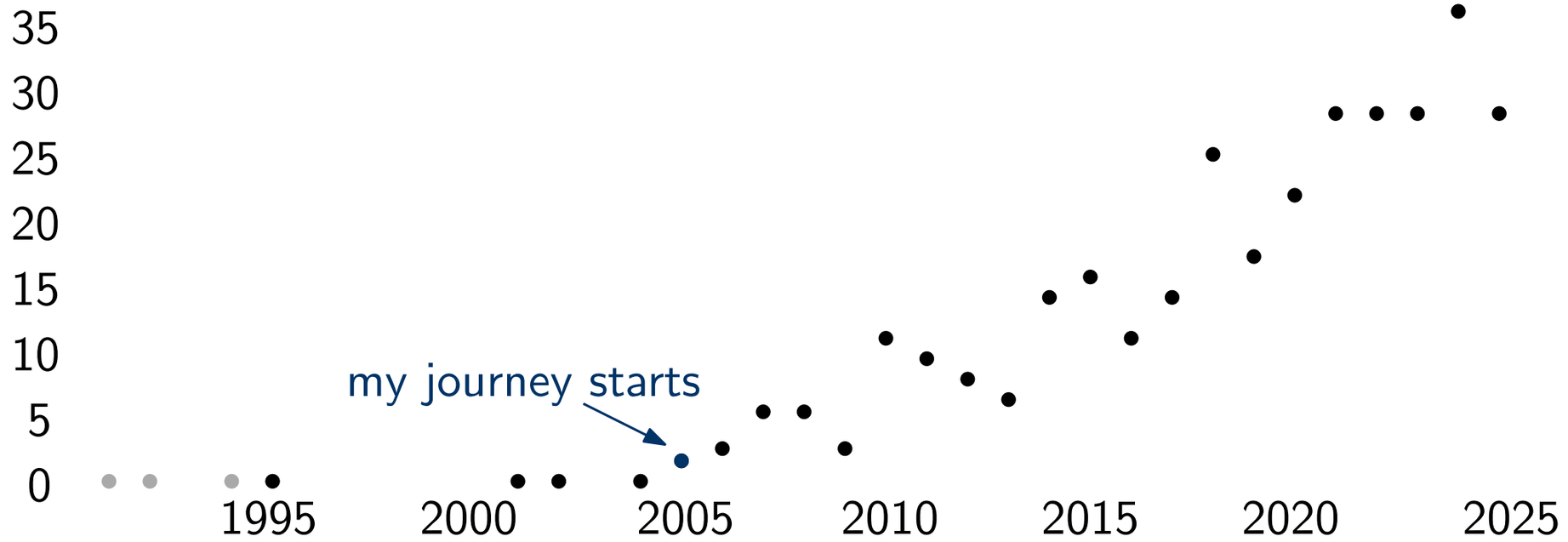




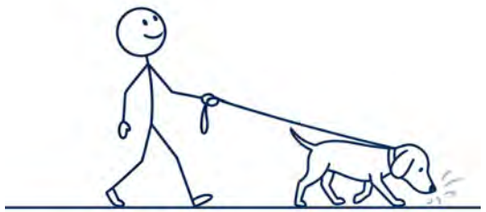
Research on computing the Fréchet distance

RUB

number of papers on dblp



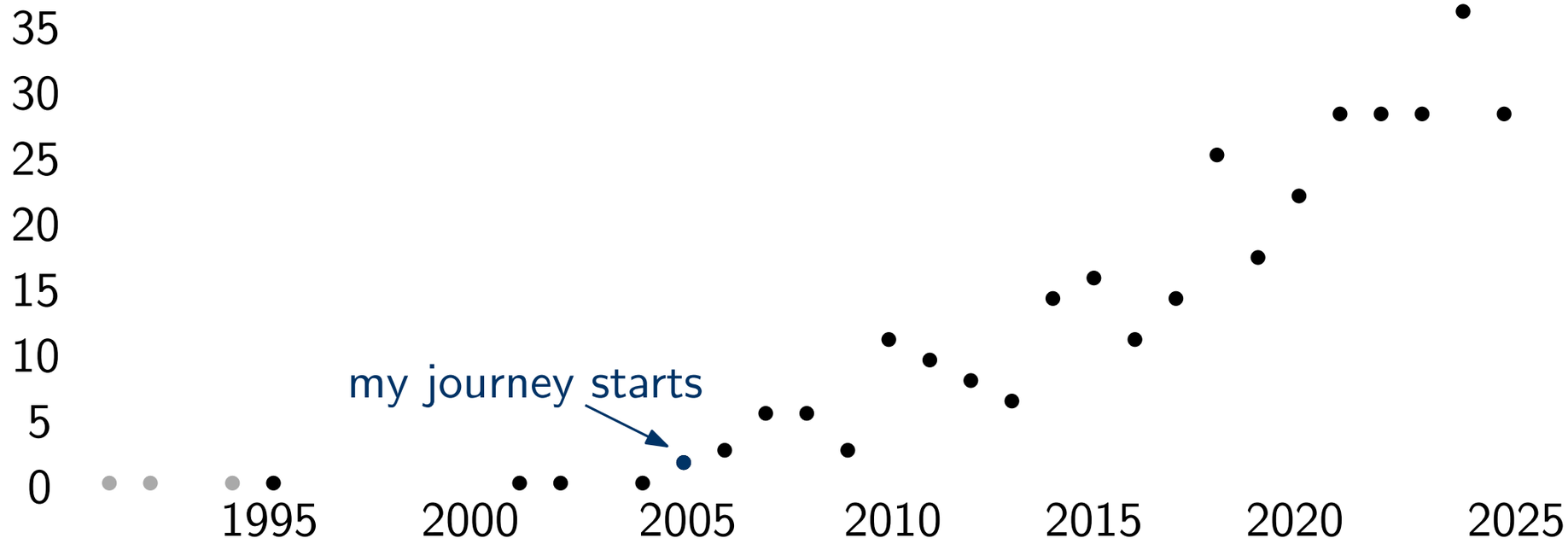
why so popular?



Research on computing the Fréchet distance

RUB

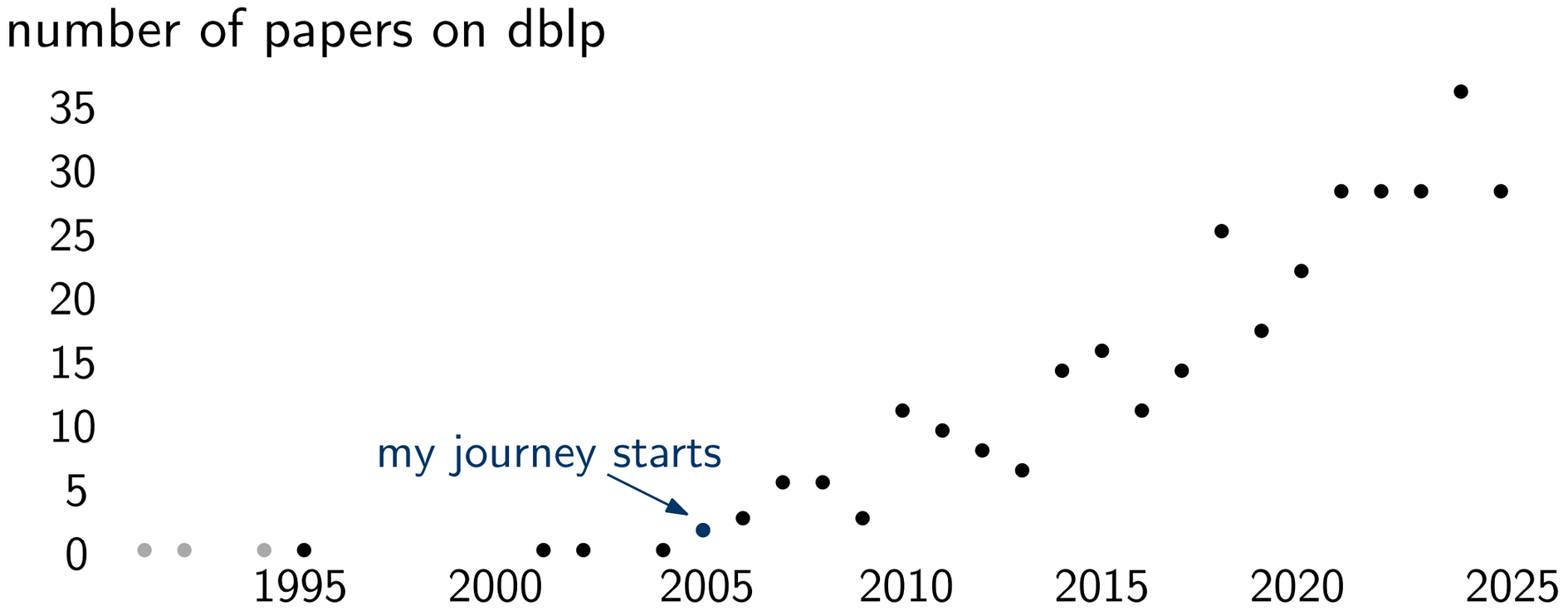
number of papers on dblp



why so popular? because it's *natural*, *versatile* and *challenging* to compute!

Research on computing the Fréchet distance

RUB



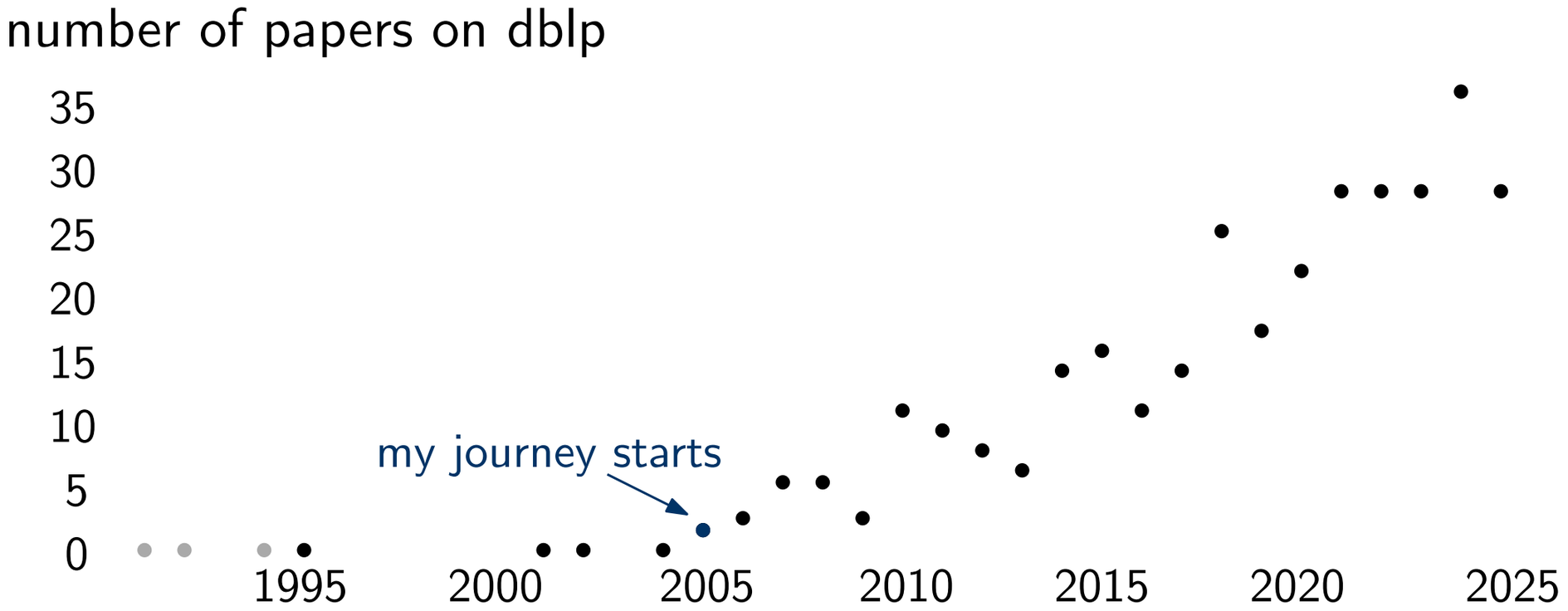
why so popular? because it's *natural*, *versatile* and *challenging* to compute!

Plan for this talk:

- some personal highlights
- open problems

Research on computing the Fréchet distance

RUB



why so popular? because it's *natural*, *versatile* and *challenging* to compute!

Plan for this talk:

- some personal highlights
- open problems

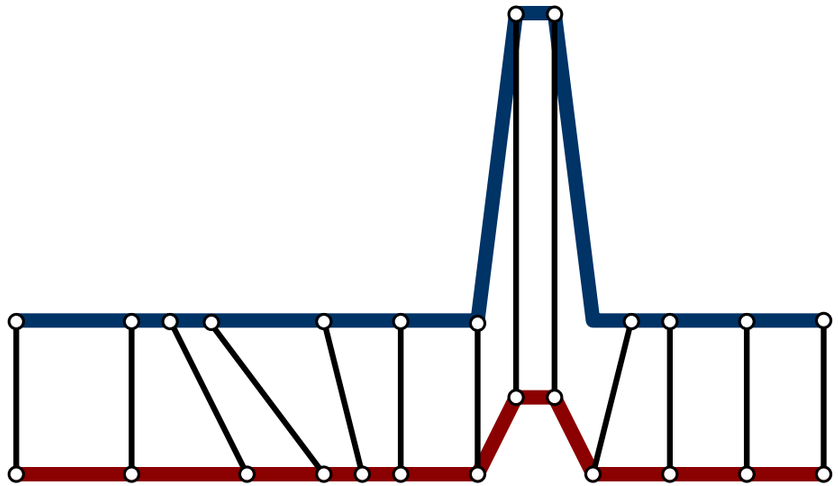
Disclaimer:
only continuous (not discrete) Fréchet distance

Algorithm of Alt & Godau

RUB

Let P, Q be two polygonal curves of size n.

- **decision problem:** Is $d_F(P, Q) \leq \varepsilon$?
- **computation problem:** What is $d_F(P, Q)$?



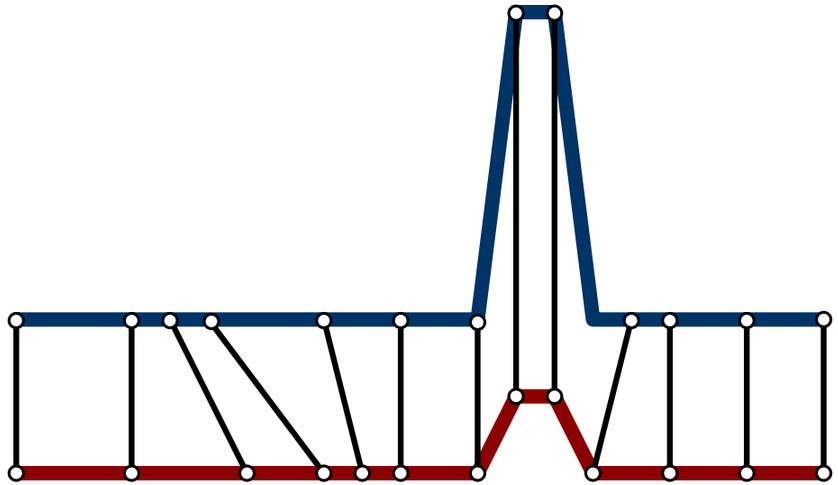
Algorithm of Alt & Godau

RUB

Let P, Q be two polygonal curves of size n.

- **decision problem:** Is $d_F(P, Q) \leq \varepsilon$?
- **computation problem:** What is $d_F(P, Q)$?

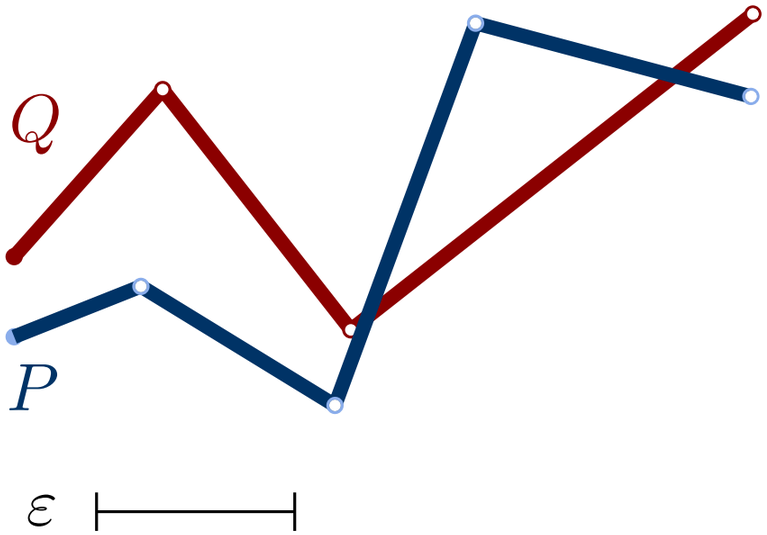
use the *Free Space Diagram* for the decision problem

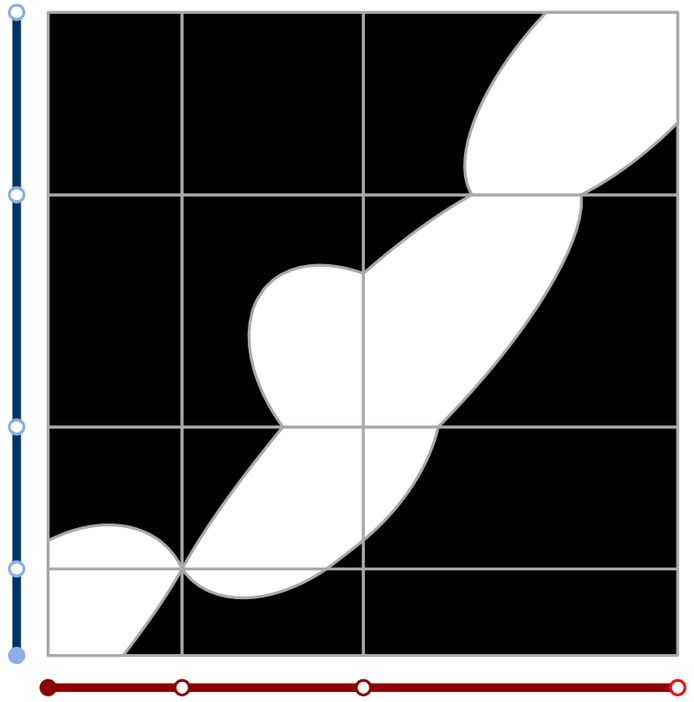


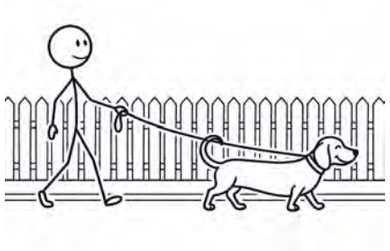
Free Space Diagram

Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$



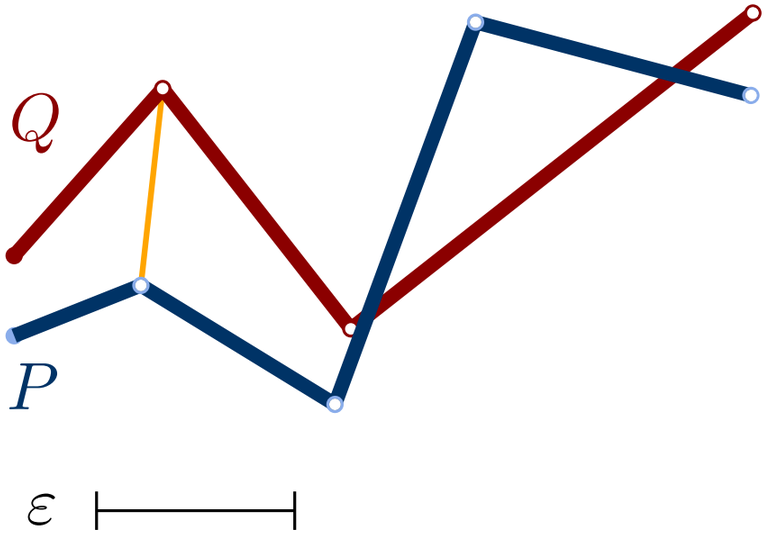


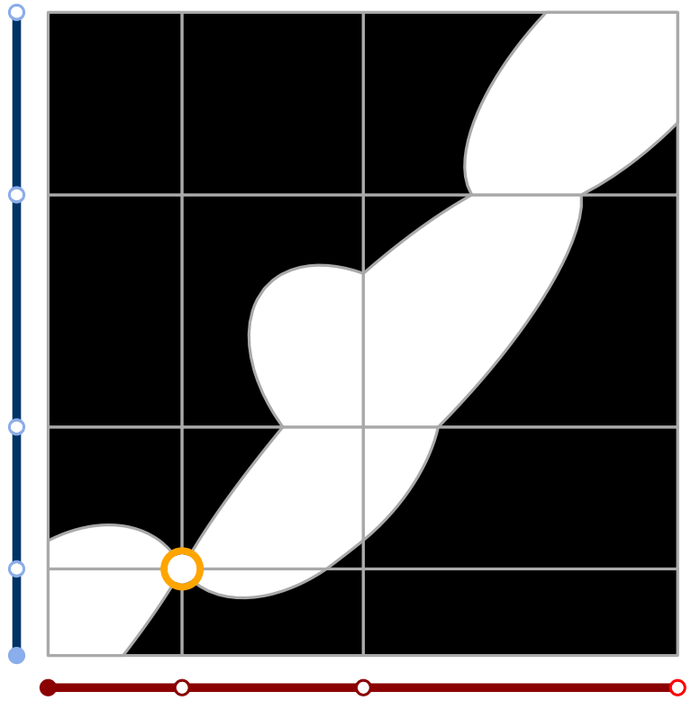


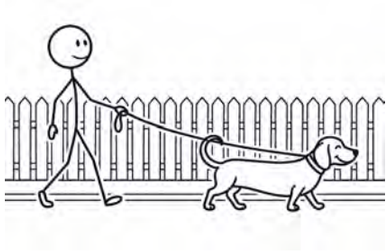
Free Space Diagram

Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$





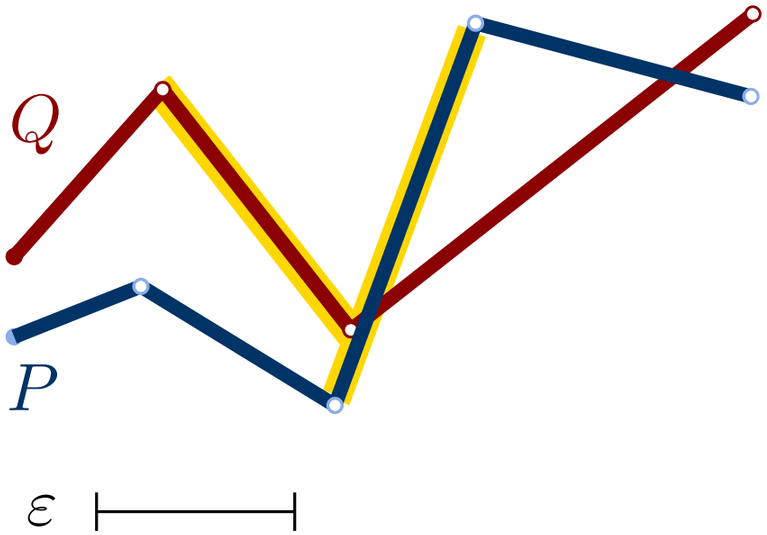


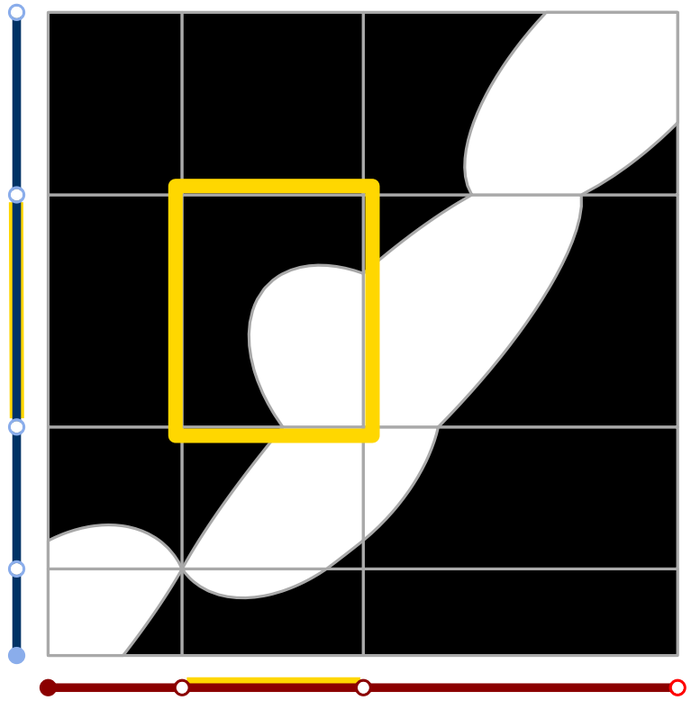
Free Space Diagram

RUB

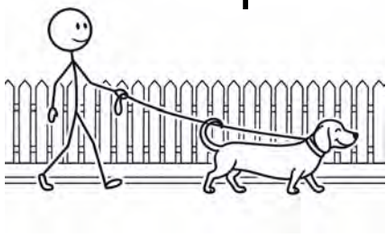
Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$





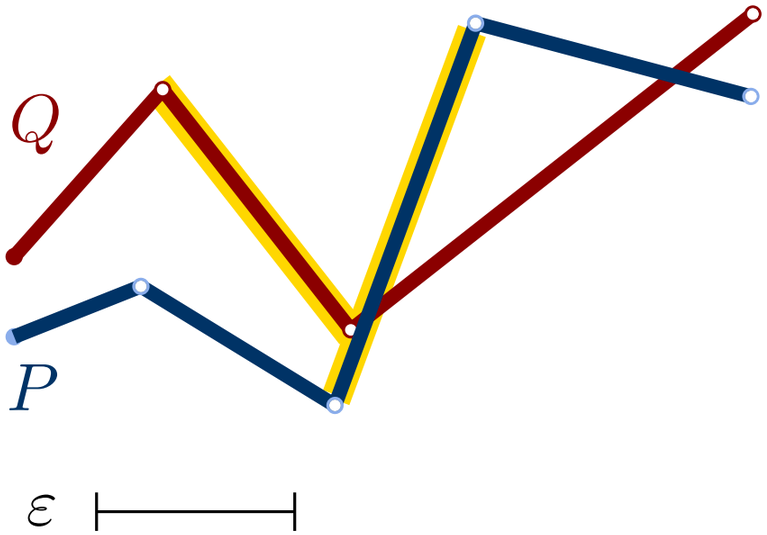
Cell: free space of two segments of the curves

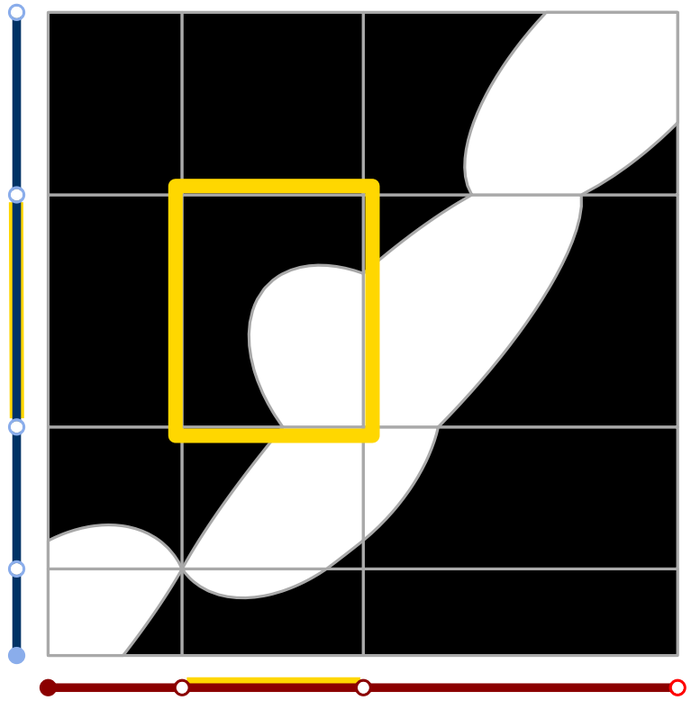


Free Space Diagram

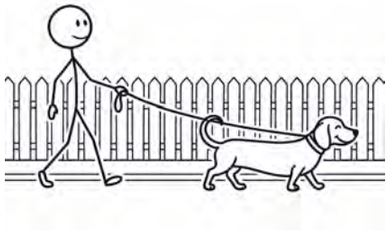
Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$





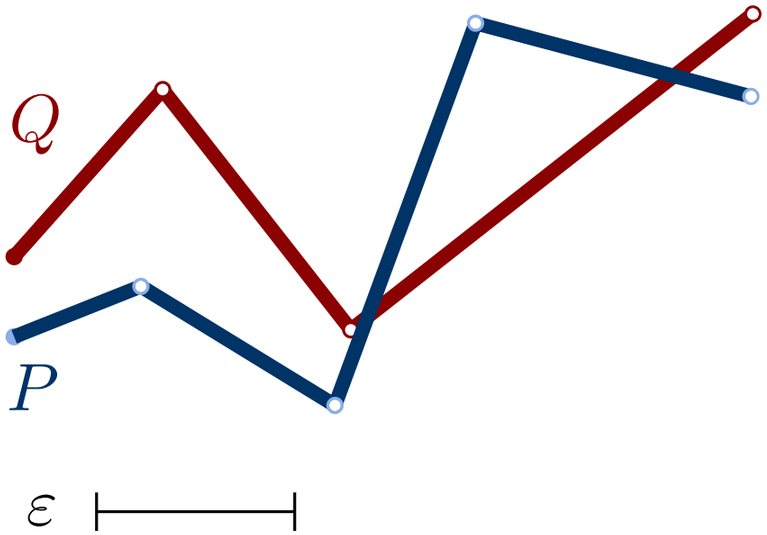
Lemma 1: Cells of the free space are convex.

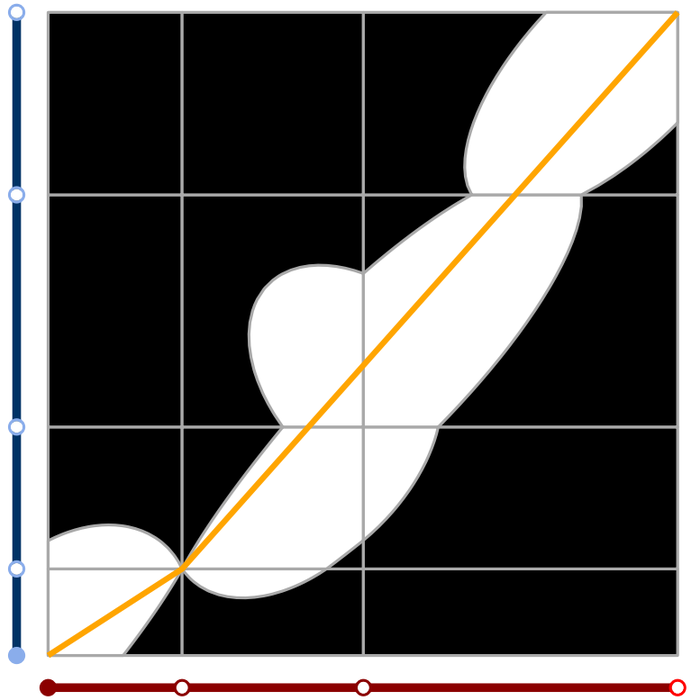


Free Space Diagram

Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$





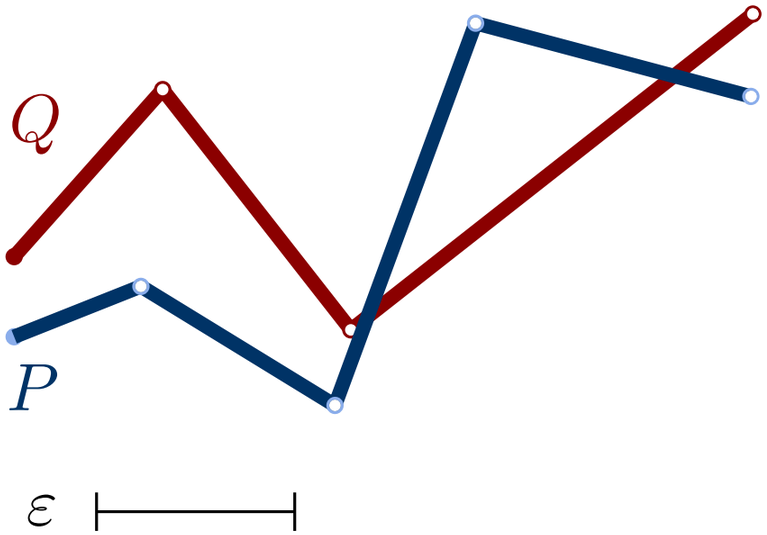
Lemma 1: Cells of the free space are convex.

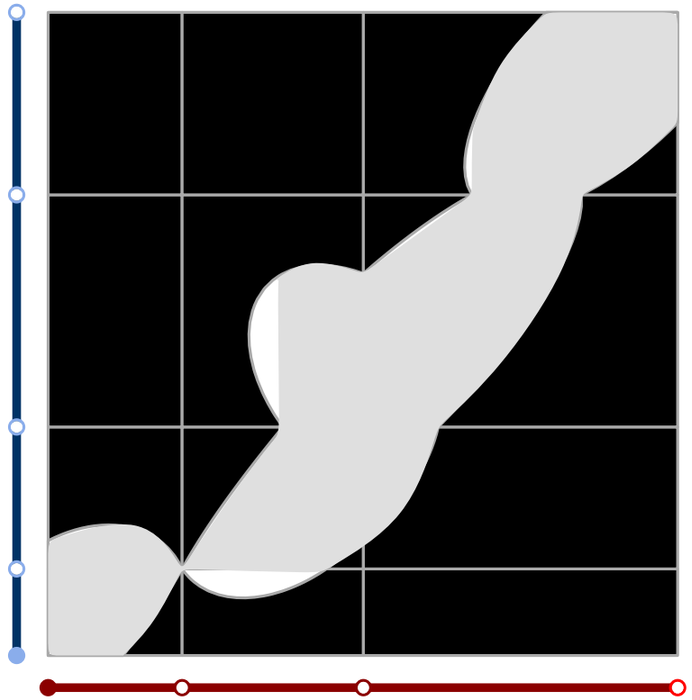
Lemma 2: $d_F(P, Q) \leq \varepsilon \Leftrightarrow$ there exists a monotone path from $(0, 0)$ to (n, n) in the free space $F_\varepsilon(P, Q)$.

Free Space Diagram

Let $P, Q\colon [0, n] \to \mathbb{R}^d$ polygonal curves, and let $\varepsilon > 0$.

Define $F_\varepsilon(P, Q) := \{(s, t) \mid d(P(s), Q(t)) \leq \varepsilon\}$



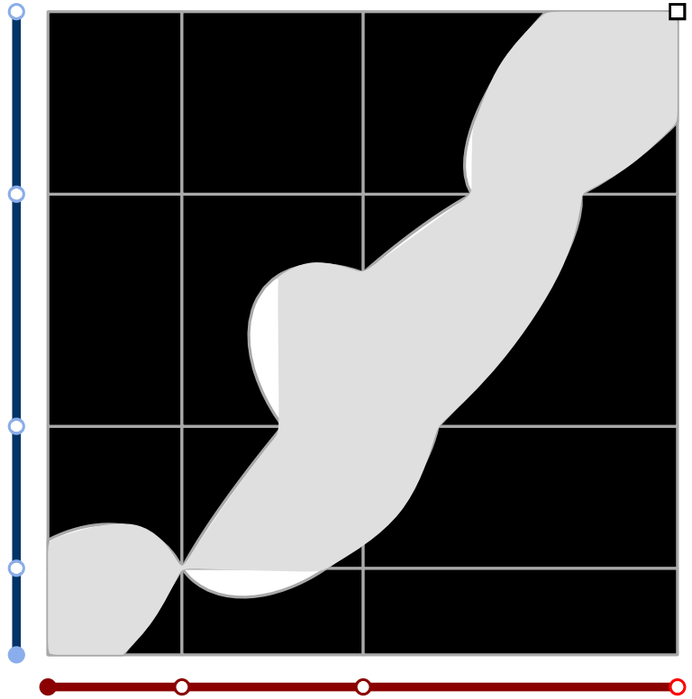


reachable free space: subset of $F_\epsilon(P, Q)$ that is reachable by a monotone path from $(0, 0)$

Decision Algorithm

Input: Polygonal curves P, Q, and $\varepsilon > 0$
Output: Is $d_F(P, Q) \leq \varepsilon$

compute lower and left reachable boundaries of the diagram
for $i \leftarrow 1$ **to** n **do**
  for $j \leftarrow 1$ **to** n **do**
    compute upper and right reachable boundaries of cell (i, j)
output **true** if (n, n) reachable, else **false**

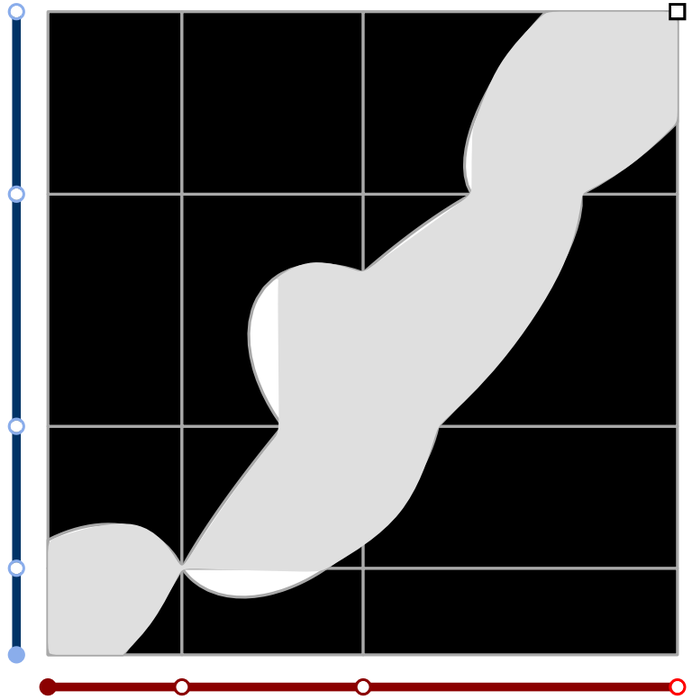


Decision Algorithm

RUB

Input: Polygonal curves P, Q, and $\varepsilon > 0$
Output: Is $d_F(P, Q) \leq \varepsilon$

compute lower and left reachable boundaries of the diagram
for $i \leftarrow 1$ **to** n **do**
  for $j \leftarrow 1$ **to** n **do**
    compute upper and right reachable boundaries of cell (i, j)
output **true** if (n, n) reachable, else **false**

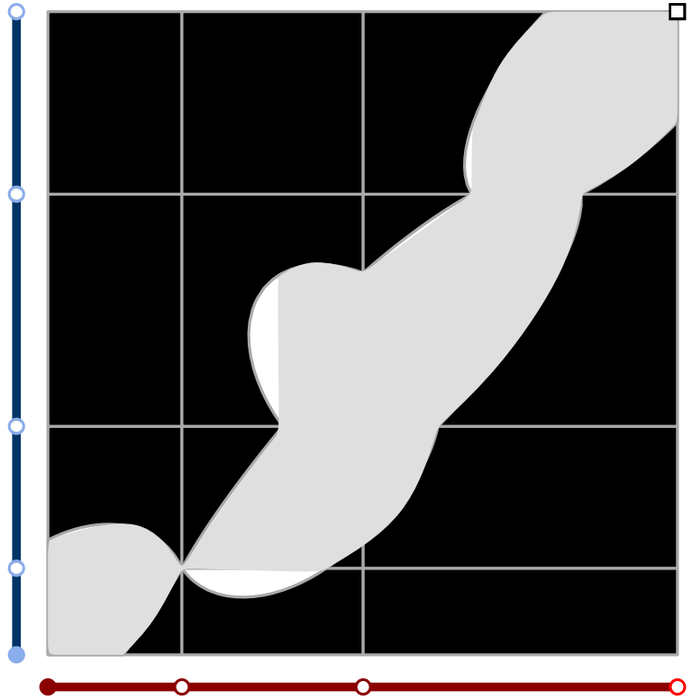


Theorem (Alt, Godau)
For two polygonal curves P, Q of size n and $\varepsilon > 0$, $d_F(P, Q) \leq \varepsilon$ can be decided in $O(n^2)$ time and $O(n)$ space.

Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step



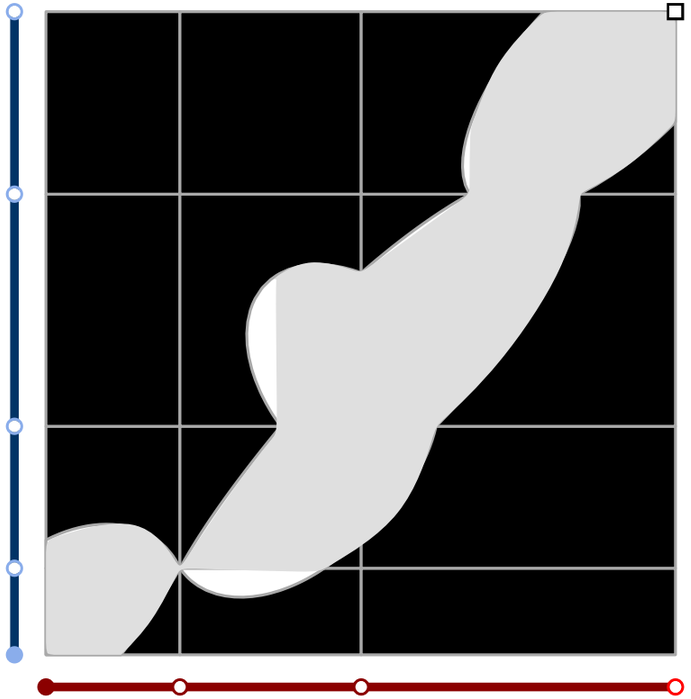
critical values:
values of ε where reachability in the free space changes

Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$



critical values:
values of ε where reachability in the free space changes

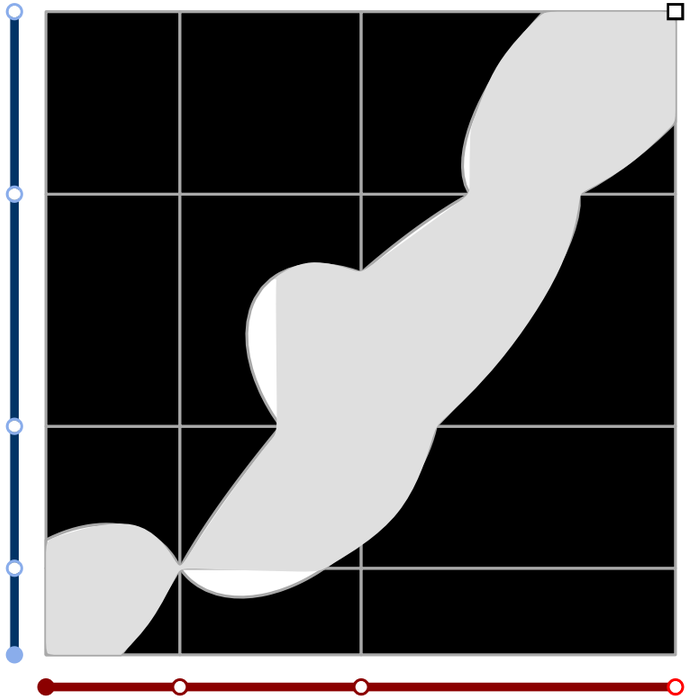
Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time



critical values:
values of ε where reachability in the free space changes

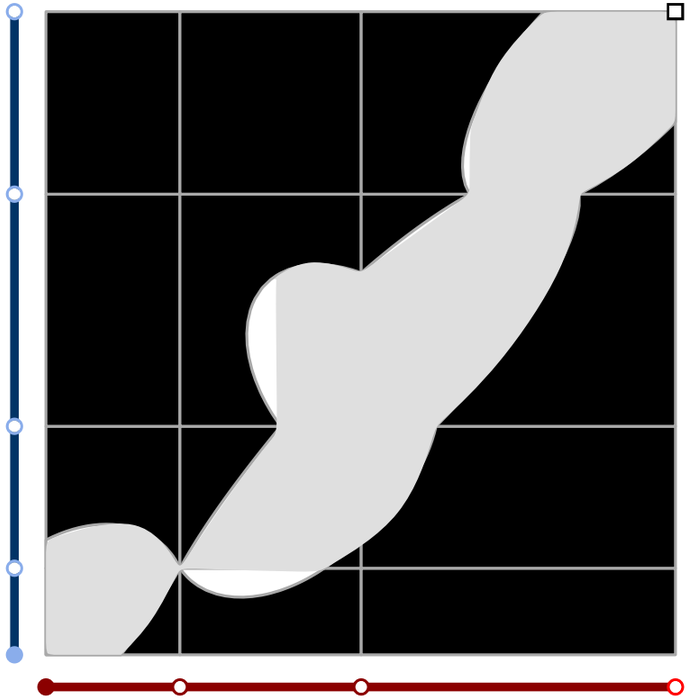
Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time



critical values:
values of ε where reachability in the free space changes

Theorem (Alt, Godau '95)
For two polygonal curves P, Q of size n, one can

- decide $d_F(P, Q) \leq \varepsilon$ in $O(n^2)$ time
- compute $d_F(P, Q)$ in $O(n^2 \log n)$ time

Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time

Theorem (Alt, Godau '95)
For two polygonal curves P, Q of size n, one can

- decide $d_F(P, Q) \leq \varepsilon$ in $O(n^2)$ time
- compute $d_F(P, Q)$ in $O(n^2 \log n)$ time

Questions:

- faster?
- implementation?
- generalizations?
- applications?

Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time

Theorem (Alt, Godau '95)
For two polygonal curves P, Q of size n, one can

- decide $d_F(P, Q) \leq \varepsilon$ in $O(n^2)$ time
- compute $d_F(P, Q)$ in $O(n^2 \log n)$ time

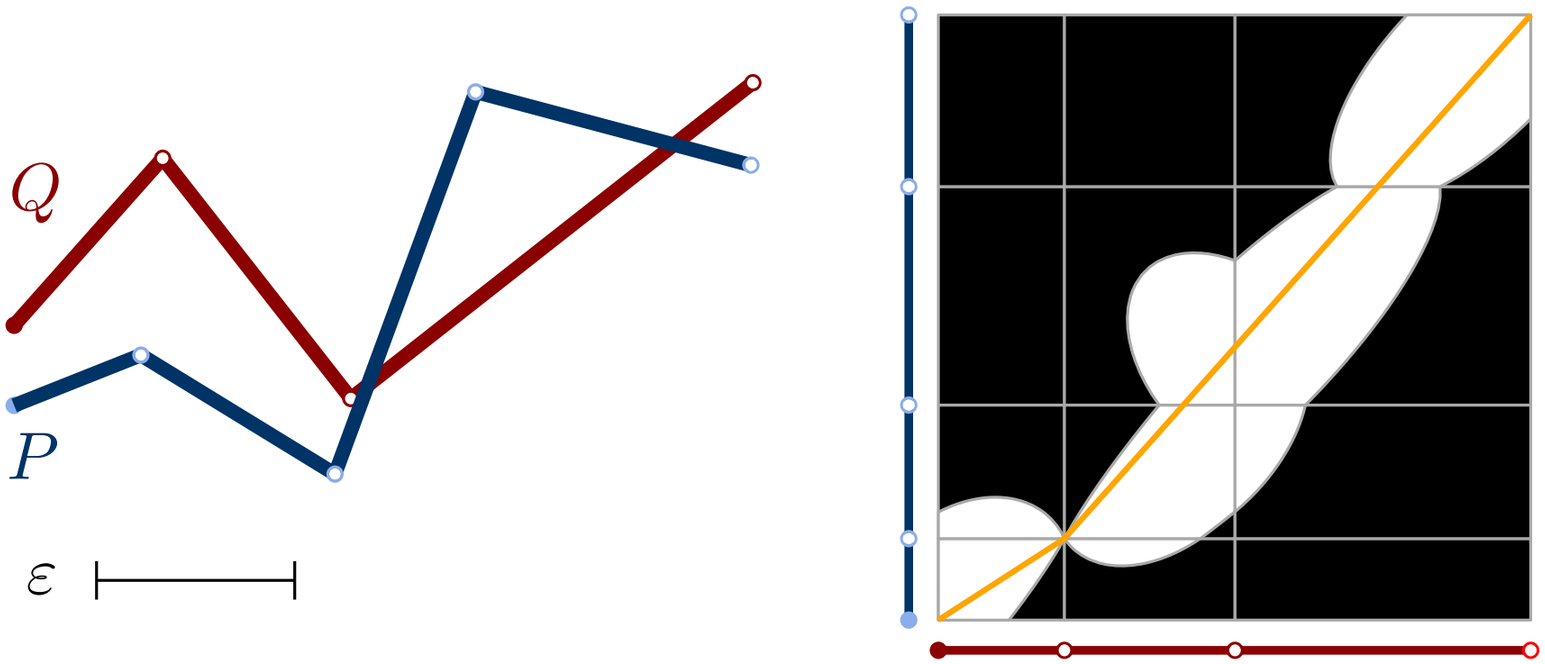
Questions:

- **faster?**
- implementation?
- generalizations?
- applications?

Breaking the quadratic runtime

RUB

Can the Fréchet distance be computed in subquadratic time?



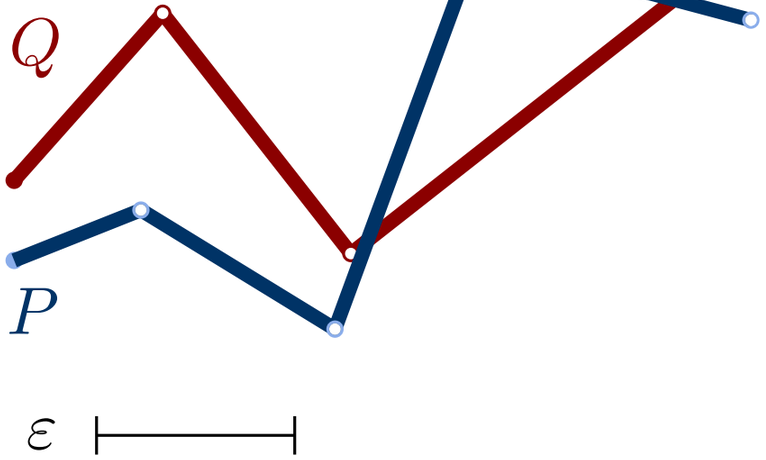
Bottleneck: size of the free space diagram

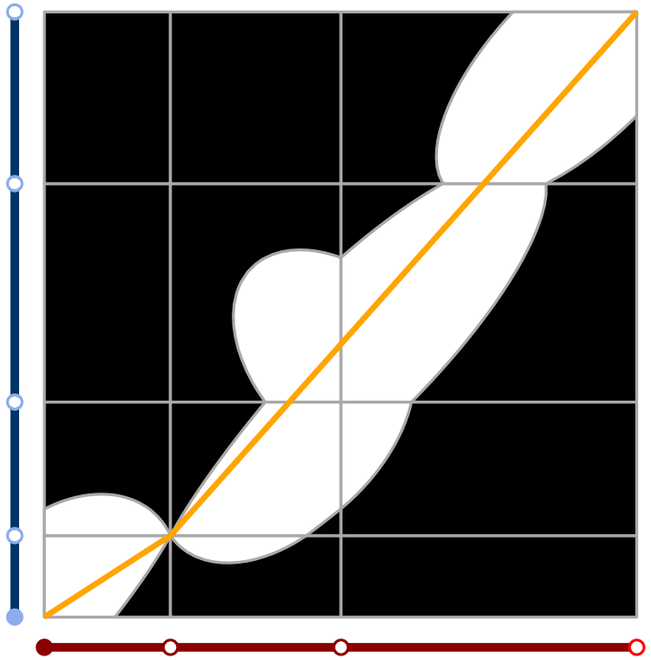
Breaking the quadratic runtime

RUB

Can the Fréchet distance be computed in subquadratic time?

- conditional lower bound by Bringmann (2014)
- slightly faster algorithm by Buchin et al. (2014)
- weakly subquadratic time by Cheng and Huang (2025)





Bottleneck: size of the free space diagram

Breaking the quadratic runtime

RUB

Can the Fréchet distance be computed in subquadratic time?

- **conditional lower bound by Bringmann (2014)**
- slightly faster algorithm by Buchin et al. (2014)
- weakly subquadratic time by Cheng and Huang (2025)

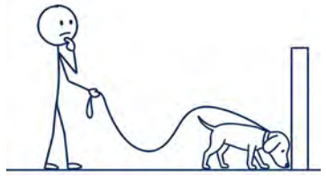
Reduction from k-CNF SAT

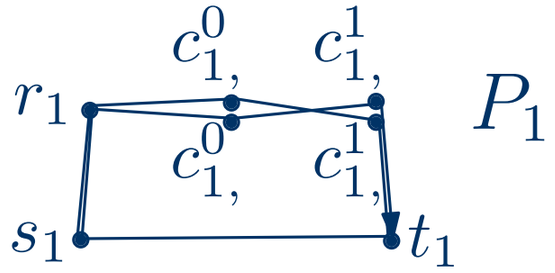
Given CNF-SAT formula F with N variables and M clauses.

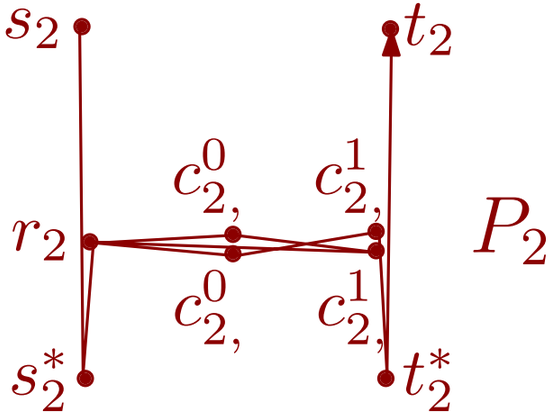
Construct two curves P_1, P_2 such that

$$d_F(P_1, P_2) \leq 1 \quad \Leftrightarrow \quad F \text{ is satisfiable.}$$

Hence an $O(n^{2-\delta})$ algorithm for d_F would imply an $O(M^2 2^{(1-\delta/2)N})$ SAT algorithm.







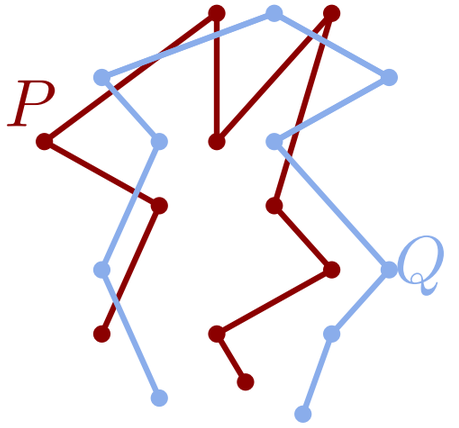
Breaking the quadratic runtime

RUB

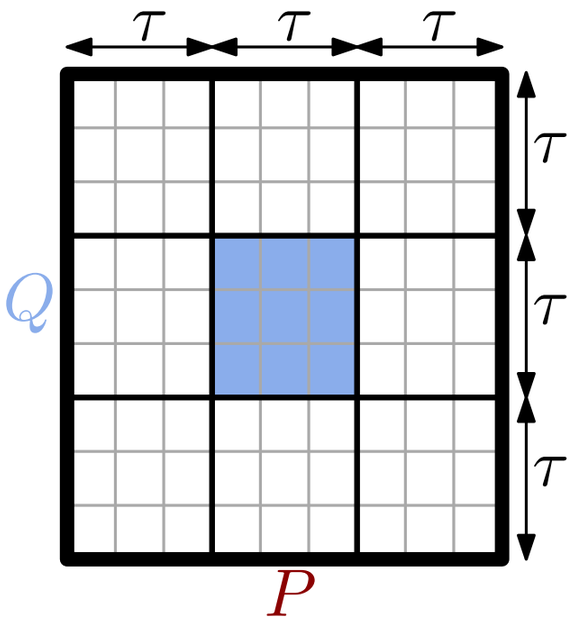
Can the Fréchet distance be computed in subquadratic time?

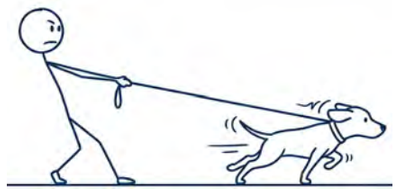
- conditional lower bound by Bringmann (2014)
- **slightly faster algorithm by Buchin et al. (2014)**
- weakly subquadratic time by Cheng and Huang (2025)

Pre-processing to speed up the computation (4 Russians trick)



elementary boxes



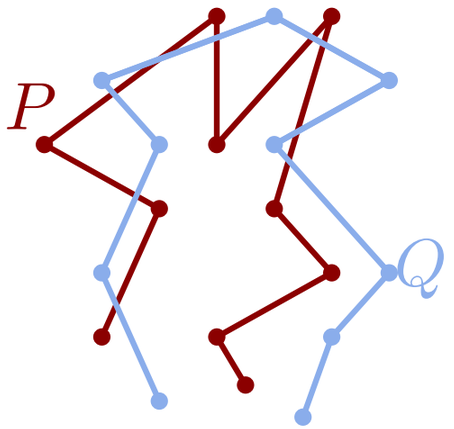


Breaking the quadratic runtime

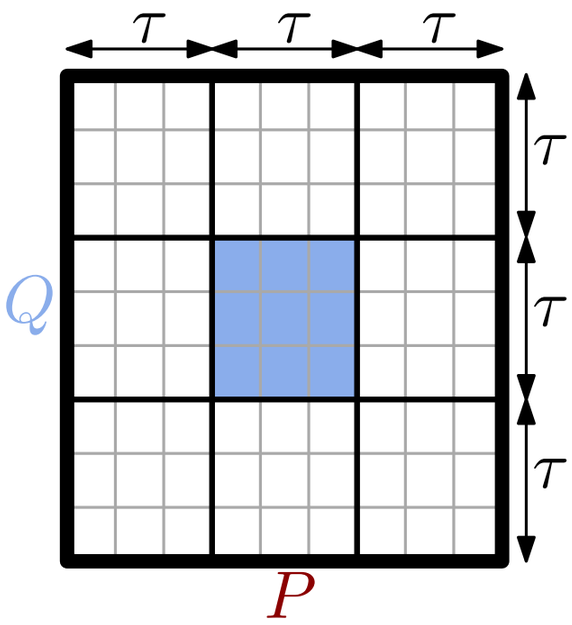
Can the Fréchet distance be computed in subquadratic time?

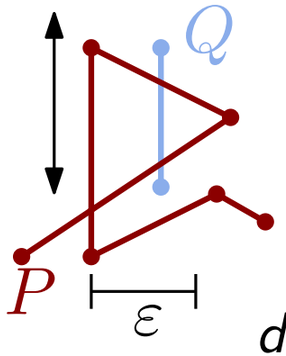
- conditional lower bound by Bringmann (2014)
- **slightly faster algorithm by Buchin et al. (2014)**
- weakly subquadratic time by Cheng and Huang (2025)

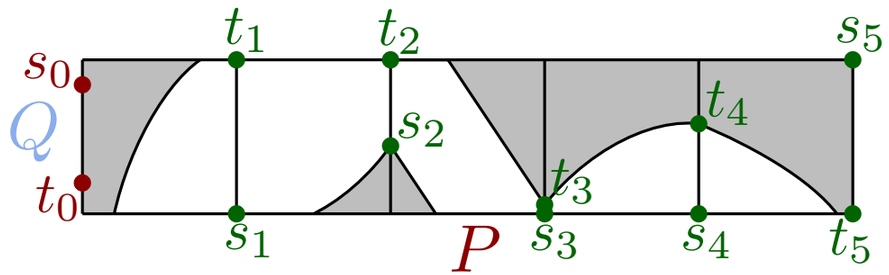
Pre-processing to speed up the computation (4 Russians trick)



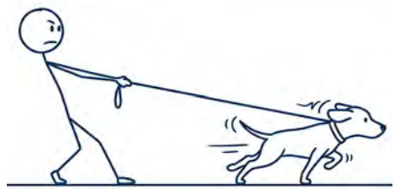
elementary boxes







door orders

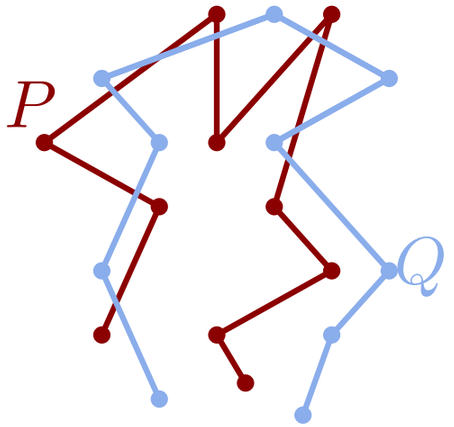


Breaking the quadratic runtime

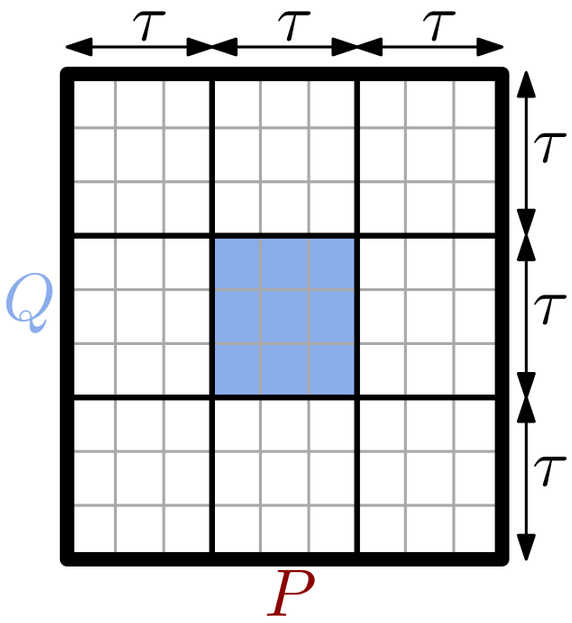
Can the Fréchet distance be computed in subquadratic time?

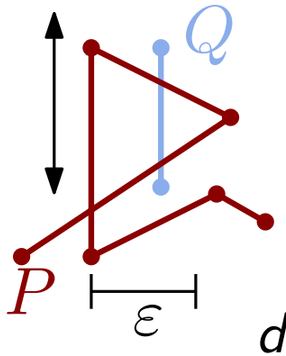
- conditional lower bound by Bringmann (2014)
- **slightly faster algorithm by Buchin et al. (2014)**
- weakly subquadratic time by Cheng and Huang (2025)

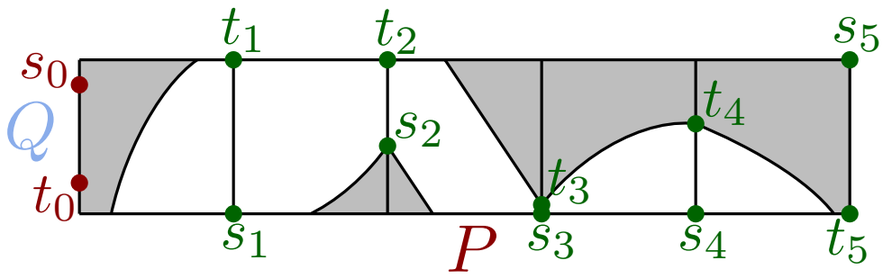
Pre-processing to speed up the computation (4 Russians trick)



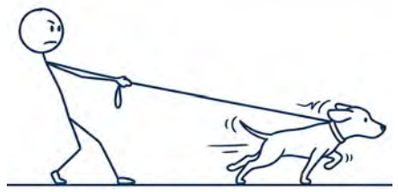
elementary boxes

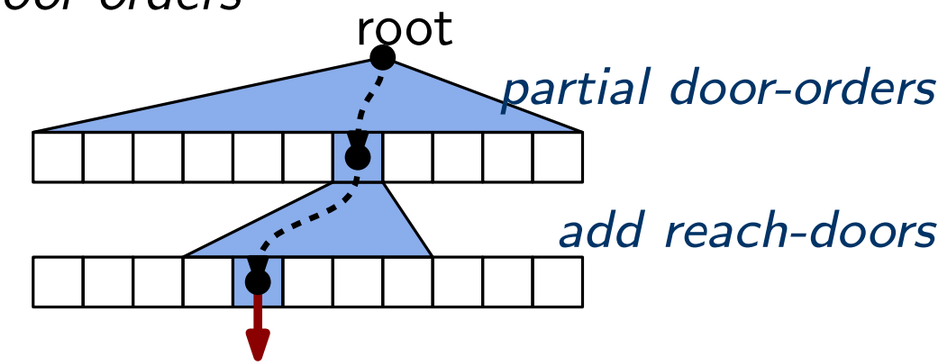






door orders





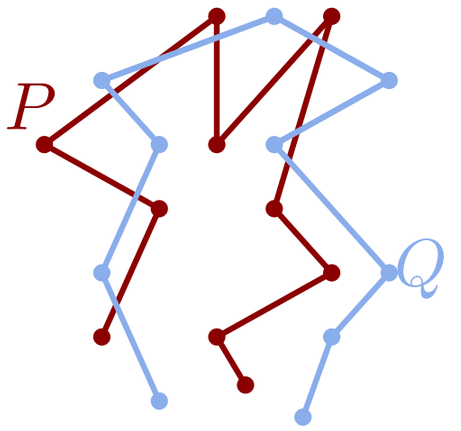
Breaking the quadratic runtime

RUB

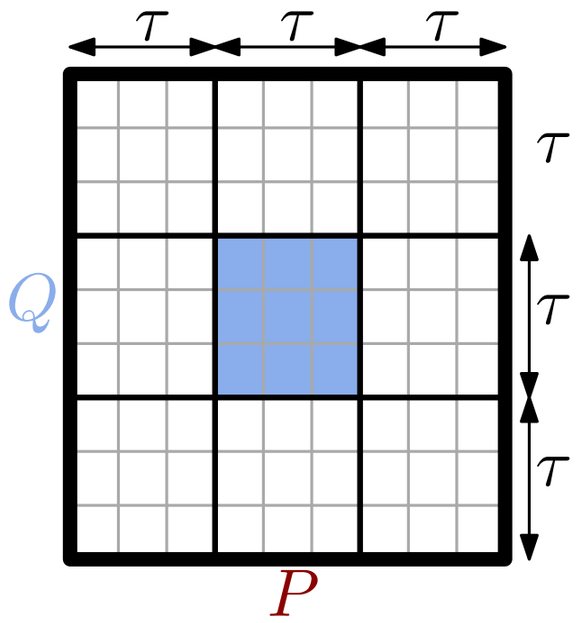
Can the Fréchet distance be computed in subquadratic time?

- conditional lower bound by Bringmann (2014)
- slightly faster algorithm by Buchin et al. (2014)
- **weakly subquadratic time by Cheng and Huang (2025)**

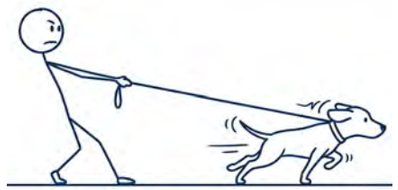
Pre-processing to speed up the computation (4 Russians trick)



elementary boxes



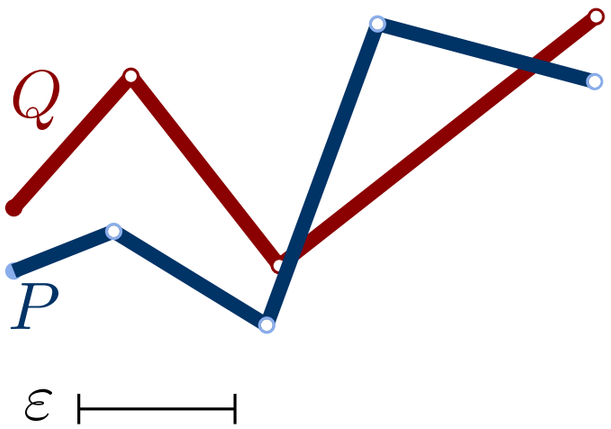
use algebraic geometric tools to show only a subquadratic number of distinct encodings

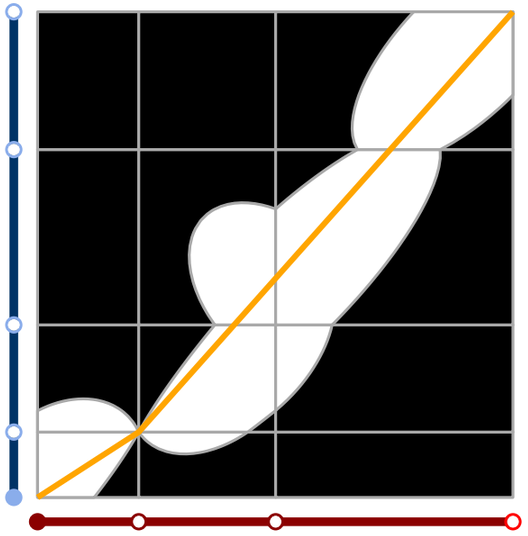


Realistic inputs

RUB

Can the Fréchet distance be computed faster for certain inputs?





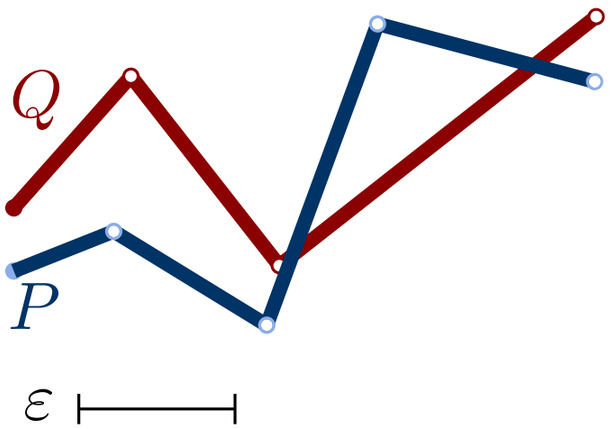
Bottleneck: size of the free space diagram

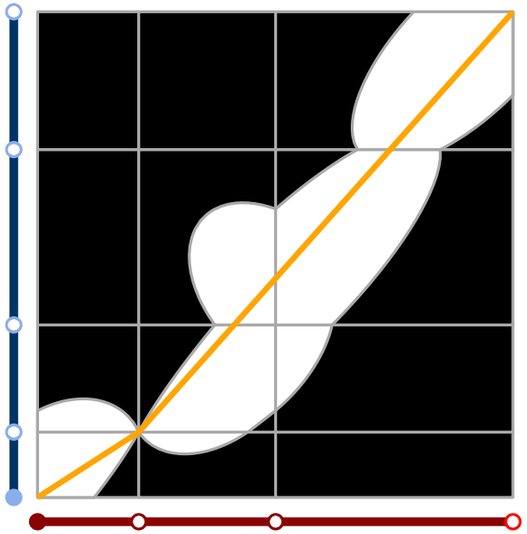
Realistic inputs

RUB

Can the Fréchet distance be computed faster for certain inputs?

- κ-straight, κ-bounded (Alt, Knauer, Wenk, 2004)
- backbone curves (Aronov et al. 2006)
- c-packed curves (Driemel, Har-Peled, Wenk, 2012),





Bottleneck: size of the free space diagram

Realistic inputs

RUB

Can the Fréchet distance be computed faster for certain inputs?

- κ-**straight, κ-bounded (Alt, Knauer, Wenk, 2004)**
- backbone curves (Aronov et al. 2006)
- c-packed curves (Driemel, Har-Peled, Wenk, 2012)

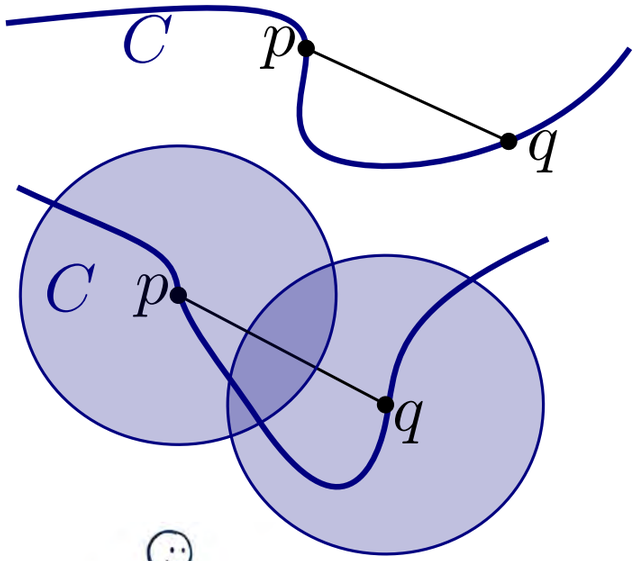
κ-straight

for p, q on curve C: $|C[p,q]| \leq \kappa||p-q||$

κ-bounded

for p, q on curve C: $C[p,q] \subseteq B(p,r) \cup B(q,r)$,
where $r = \kappa||p-q||/2$

for these curves Fréchet distance can be
bounded by $\kappa + 1$ times Hausdorff distance



Realistic inputs

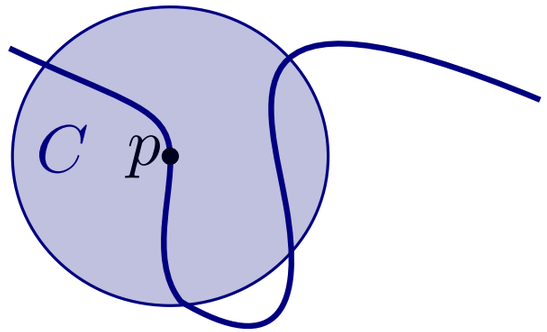
RUB

Can the Fréchet distance be computed faster for certain inputs?

- κ-straight, κ-bounded (Alt, Knauer, Wenk, 2004)
- backbone curves (Aronov et al. 2006)
- **c-packed curves (Driemel, Har-Peled, Wenk, 2012)**

c-packed
for any p: $|C \cap B(p,r)| \leq cr$

- simplify curves (c-packed remains $6c$-packed, complexity of fsd reduced)
- approximately decide Fréchet distance between simplifications



Realistic inputs

RUB

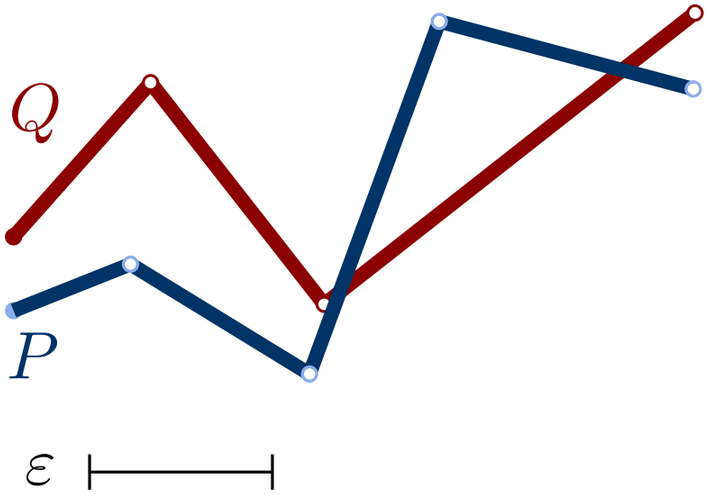
Can the Fréchet distance be computed faster for certain inputs?

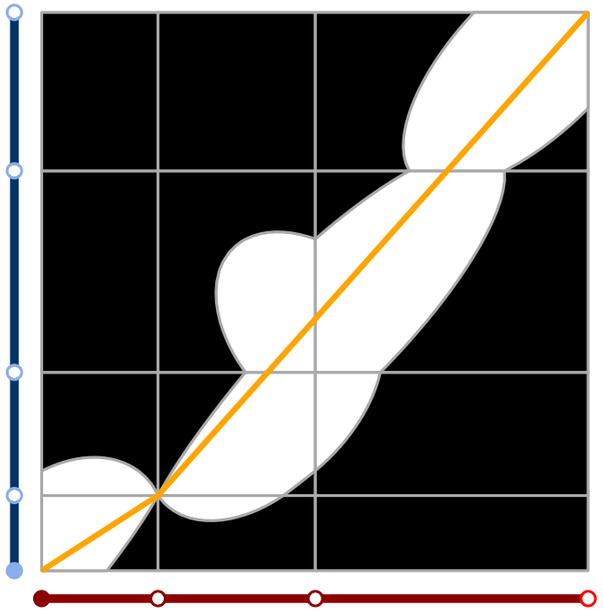
- κ-straight, κ-bounded (Alt, Knauer, Wenk, 2004)
- backbone curves (Aronov et al. 2006)
- c-packed curves (Driemel, Har-Peled, Wenk, 2012),
- long edges (Gudmundsson, Mirzanezhad, Mohades, Wenk, 2019)
- one curve c-packed (Gudmundsson, Mai, Wong, 2024),
  (Conradi, van der Hoog, van der Horst, Ophelders, 2026)
- imbalanced (Blank, 2026)

Approximation

RUB

Can the Fréchet distance be approximated faster?



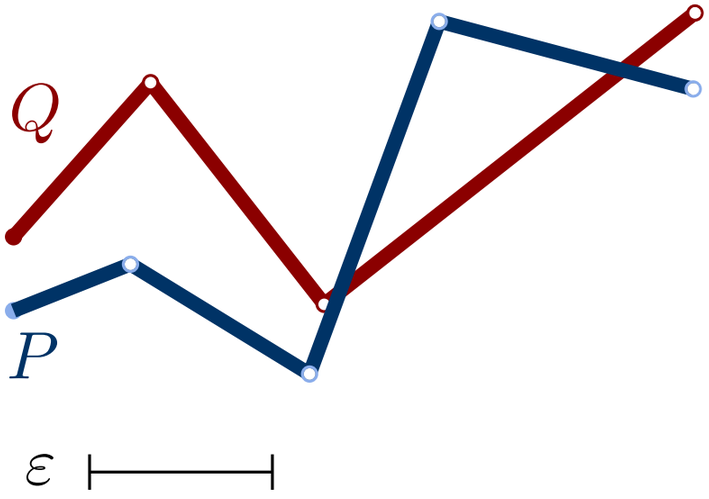


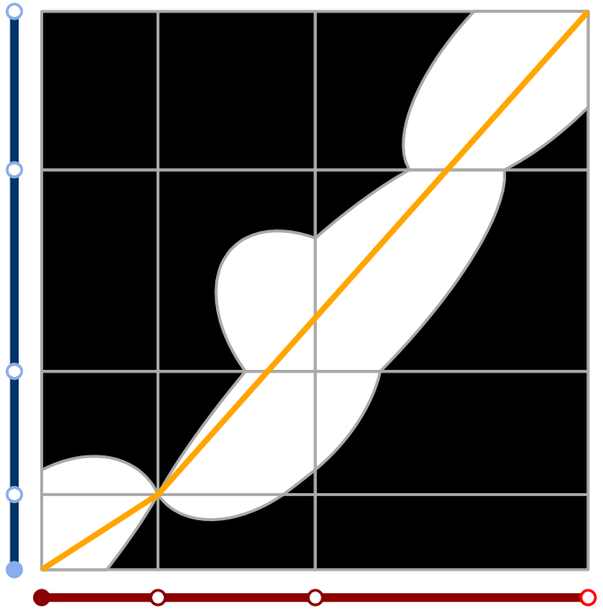
Approximation

RUB

Can the Fréchet distance be approximated faster?

- no 3-approx in subquadratic time (Buchin, Ophelders, Speckmann, 2019)
- $O(\alpha)$-approx for $\alpha \in [\sqrt{n}, n]$ in $O((n^3/\alpha^2)\log^3 n)$ (Colombe & Fox, 2021)
- $O(\alpha)$-approx for $\alpha \in [1, n]$ in $O((n + mn/\alpha)\log^3 n)$ (van der Horst, van Kreveld, Ophelders & Speckmann, 2023)
- $O(\alpha)$-approx for $\alpha \in [1, n]$ in $O((n + mn/\alpha)\log^2 n)$ (van der Horst, Ophelders, '24)
- $(7 + \varepsilon)$-approx in subquadratic time (Cheng, Huang & Zhang, 2025)



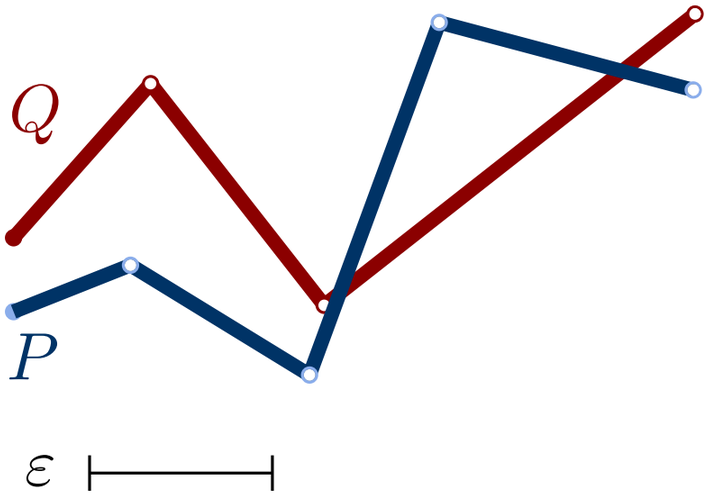


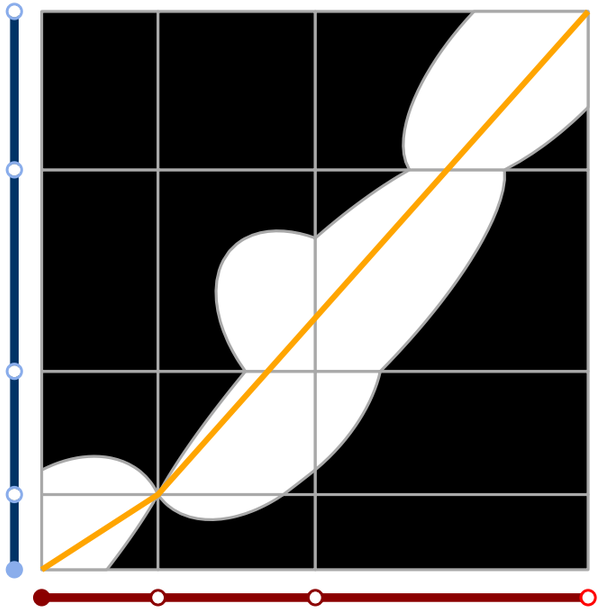
Approximation

RUB

Can the Fréchet distance be approximated faster?

- no 3-approx in subquadratic time (Buchin, Ophelders, Speckmann, 2019)
- $O(\alpha)$-**approx for** $\alpha \in [\sqrt{n}, n]$ **in** $O((n^3/\alpha^2)\log^3 n)$ **(Colombe & Fox, 2021)**
- $O(\alpha)$-approx for $\alpha \in [1, n]$ in $O((n + mn/\alpha)\log^3 n)$ (van der Horst, van Kreveld, Ophelders & Speckmann, 2023)
- $O(\alpha)$-approx for $\alpha \in [1, n]$ in $O((n + mn/\alpha)\log^2 n)$ (van der Horst, Ophelders, '24)
- $(7 + \varepsilon)$-approx in subquadratic time (Cheng, Huang & Zhang, 2025)



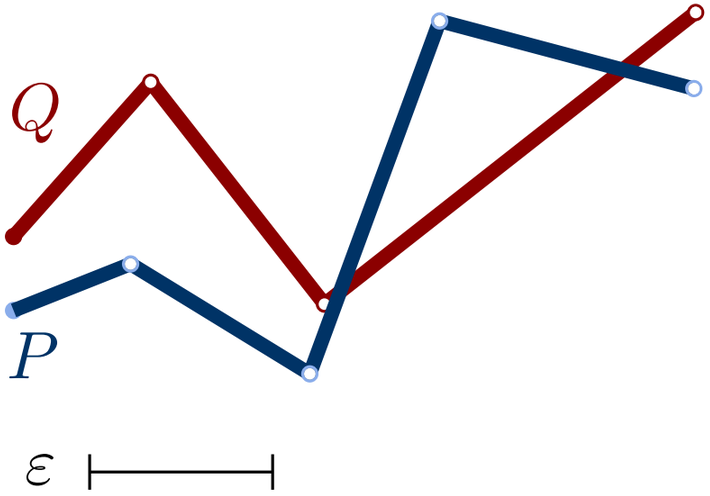


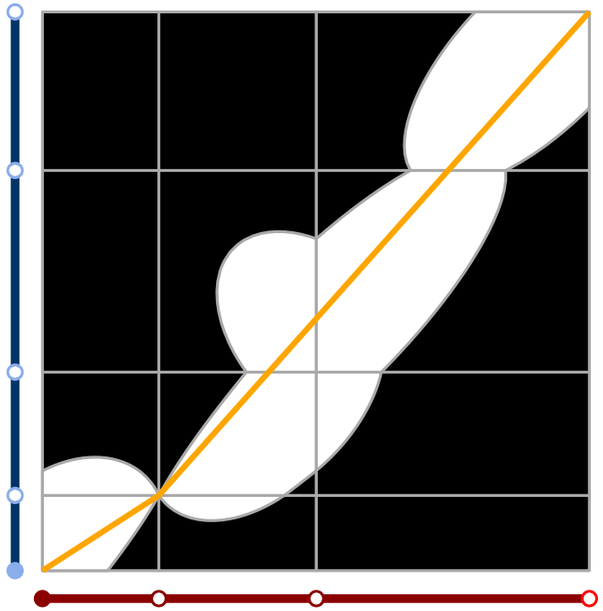
combines

- simplification
- grids
- greedy (for long edges)

Implementation

How to efficiently implement the algorithms?





Implementation

RUB

How to efficiently implement the algorithms?

ACM SIGSPATIAL Cup 2017

Home | Results | Problem Definition | Downloads | Submission and Evaluation | Organizing Committee

Trajectory similarity is a very basic problem when dealing with this type of data and many different ways of comparing trajectories have been proposed. In this challenge, we concentrate on a very intuitive measure called **Fréchet distance** with hard computational complexity, but superior quality in theory and practice.

Paper of the Winning Teams

All winning teams were invited to publish a paper describing their approach in the proceedings of SIGSPATIAL GIS.

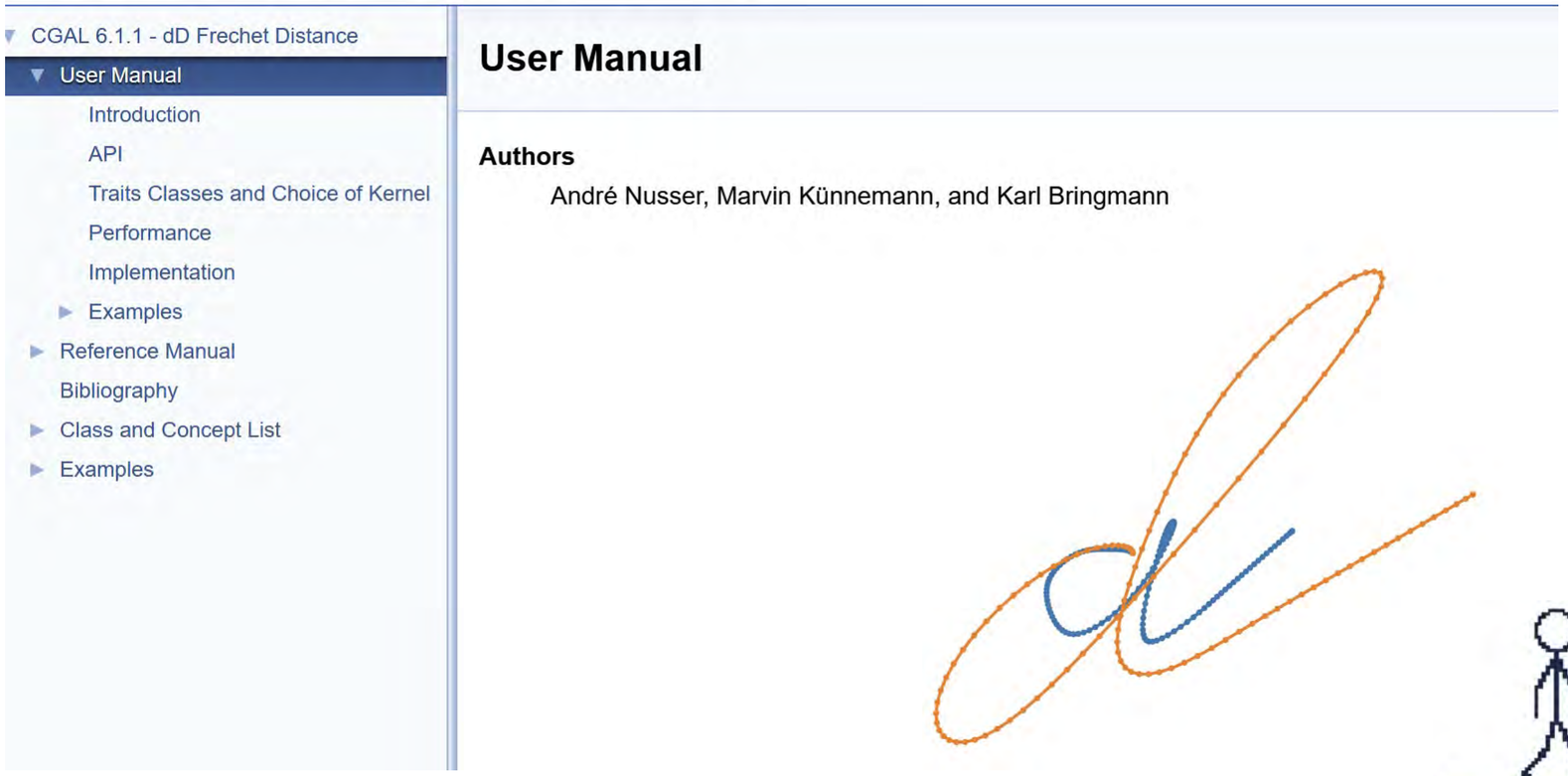
- **A fast implementation of near neighbors queries for fréchet distance.**
  J. Baldus and K. Bringmann.
- **Efficient trajectory queries under the fréchet distance.**
  K. Buchin, Y. Diez, T. van Diggelen, and W. Meulemans.
- **A filter-and-refinement-algorithm for range queries based on the fréchet distance.**
  F. Dütsch and J. Vahrenhold.

Implementation

How to efficiently implement the algorithms?

- Walking the Dog Fast in Practice: Algorithm Engineering of the Fréchet Distance (Bringmann, Künnemann, Nusser, 2019)

CGAL 6.1.1 - dD Frechet Distance



Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time

Theorem (Alt, Godau '95)
For two polygonal curves P, Q of size n, one can

- decide $d_F(P, Q) < \varepsilon$ in $O(n^2)$ time
- compute $d_F(P, Q)$ in $O(n^2 \log n)$ time

Questions:

- faster!
- implementation!
- generalizations?
- applications?

Computation Algorithm

RUB

Input: Polygonal curves P, Q
Output: $d_F(P, Q)$
determine all critical values
sort the critical values
do a **binary search** over the values,
solving the decision problem in each step

Runtime: $O(n^3 \log n)$

faster: with *parametric search* in $O(n^2 \log n)$ time

Theorem (Alt, Godau '95)
For two polygonal curves P, Q of size n, one can

- decide $d_F(P, Q) < \varepsilon$ in $O(n^2)$ time
- compute $d_F(P, Q)$ in $O(n^2 \log n)$ time

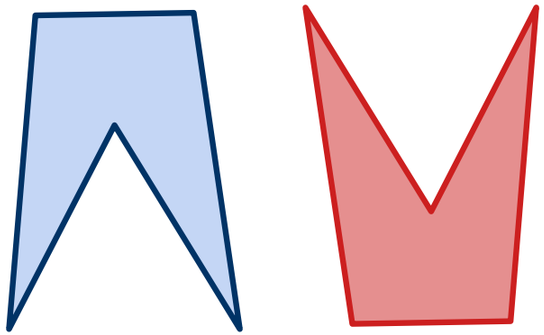
Questions:

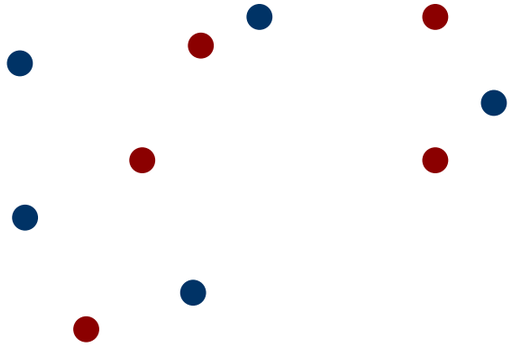
- faster!
- implementation!
- **generalizations?**
- **applications?**

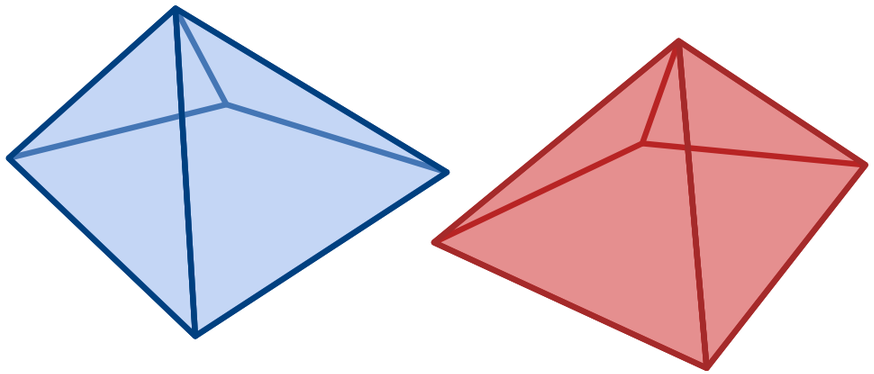
Generalizations

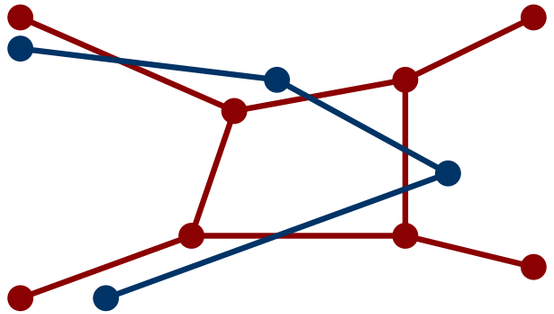
RUB

What about other shapes, e.g., surfaces, polygons, graphs, or point sets?





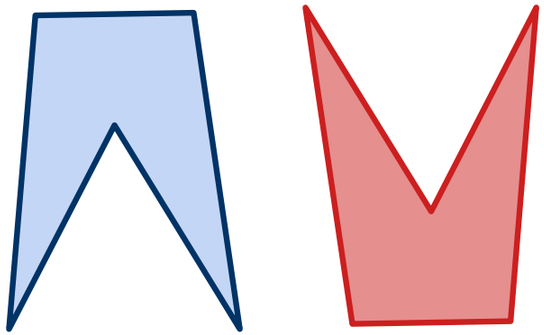


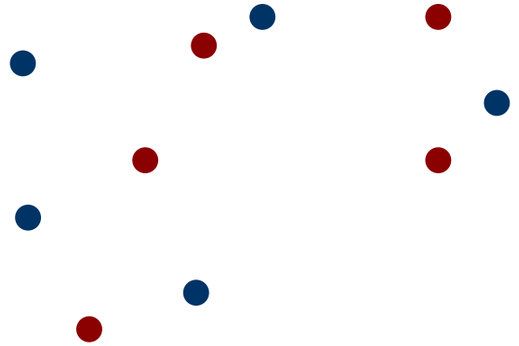


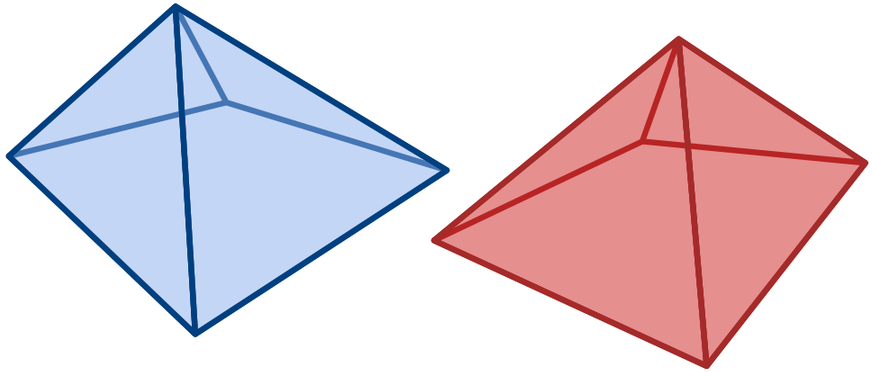
Generalizations

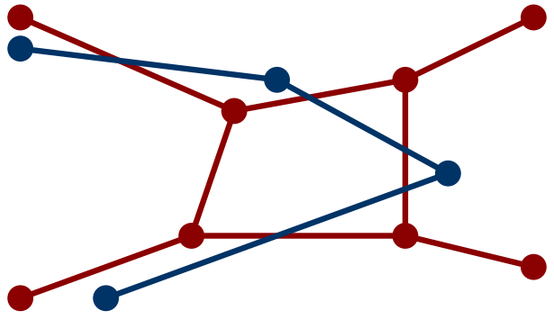
What about other shapes, e.g., surfaces, polygons, graphs, or point sets?

$\rightarrow$ Fréchet distance has been generalized to all of these









several of these turn out to be NP-hard

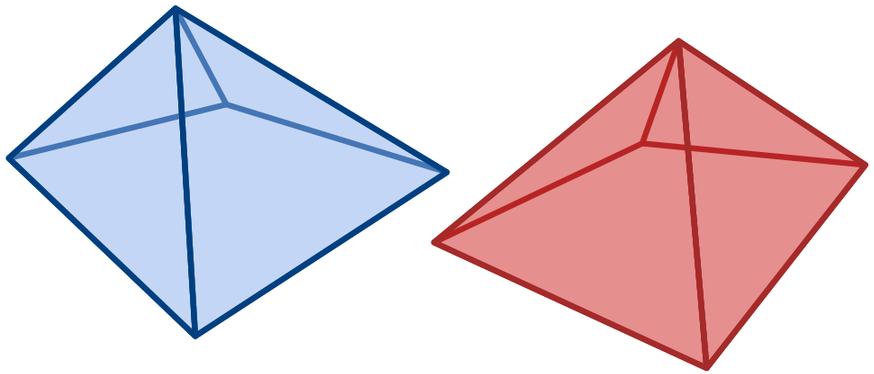
Generalizations

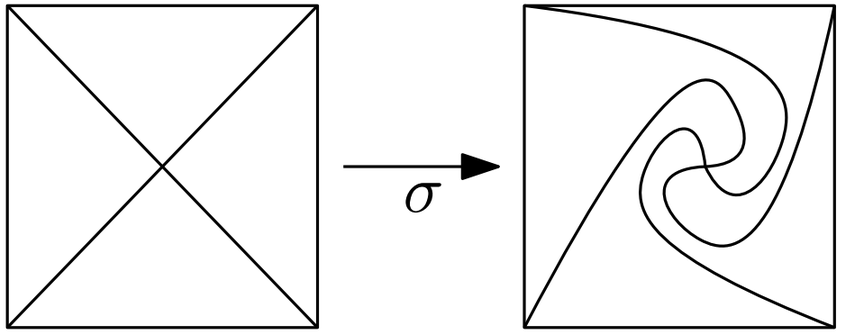
RUB

What about other shapes, e.g., **surfaces**, polygons, graphs, or point sets?

$\rightarrow$ Fréchet distance has been generalized to all of these

- generalized to surfaces (Fréchet, 1924)
- NP-hard (Godau, 1998), (Buchin, Buchin, Schulz 2010)
- Semi-computable (Alt, Buchin, 2004)
- Computable (Nayyeri, Xu, 2016)



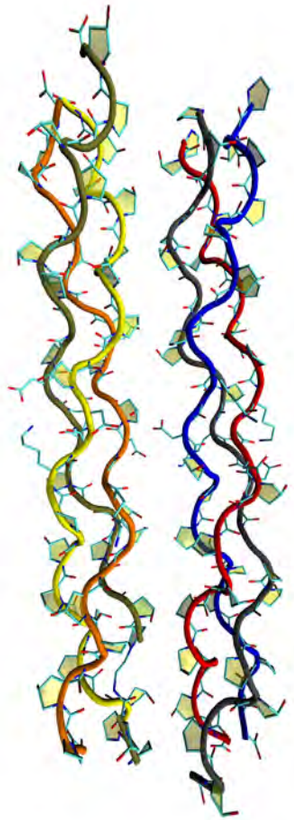


Homeomorphisms on $[0,1]^2$ are not as nice as on $[0,1]$

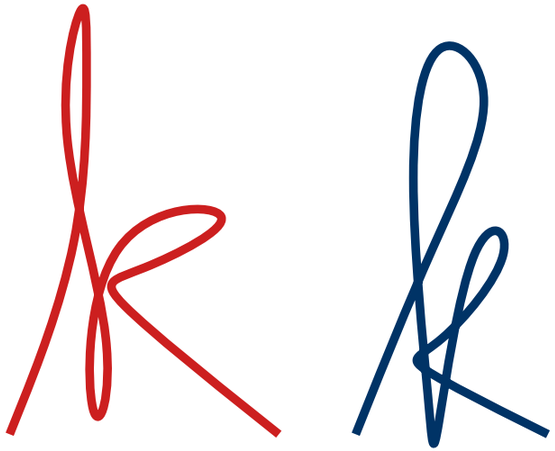
Under Transformation

RUB

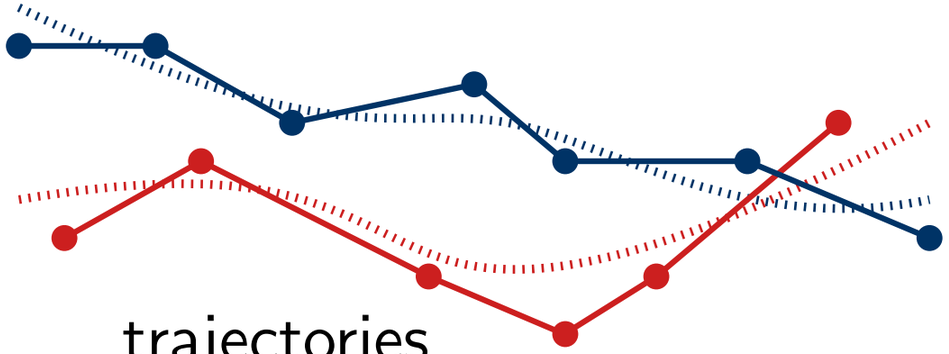
Shape matching: given two curves, find a transformation minimizing their distance.



protein backbones



handwriting



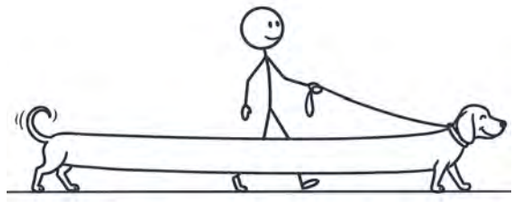
trajectories

Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)

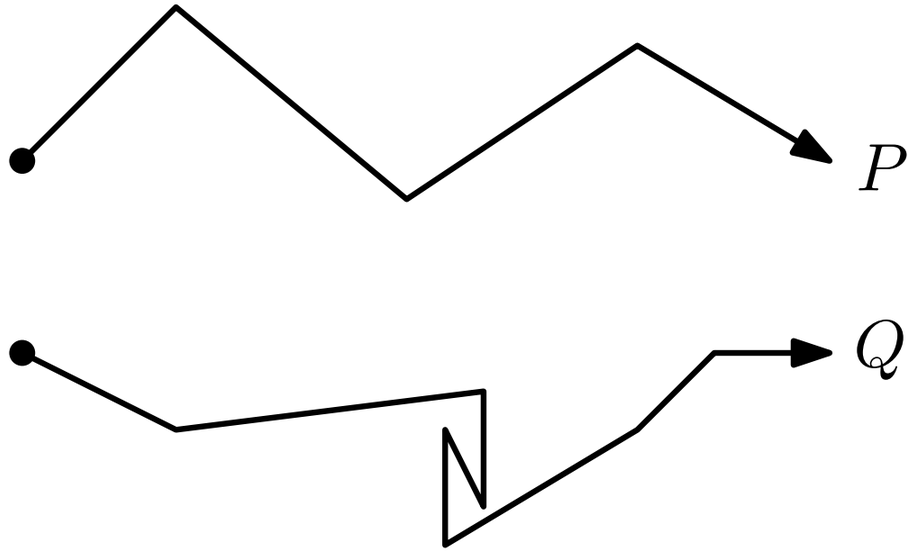


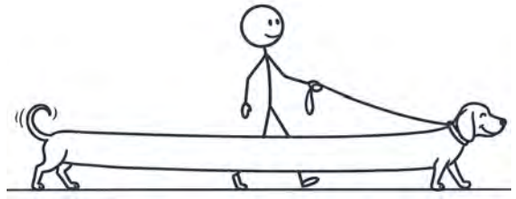
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)



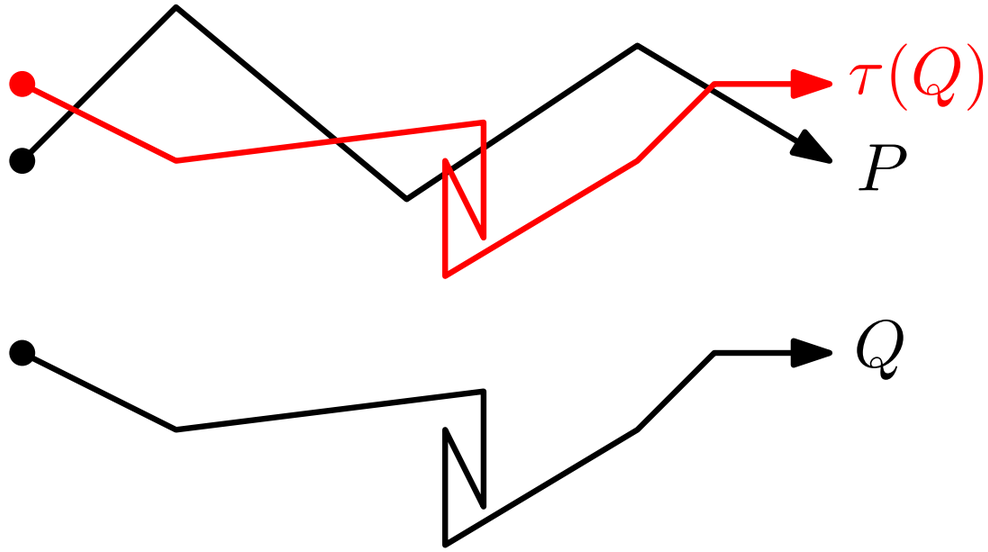


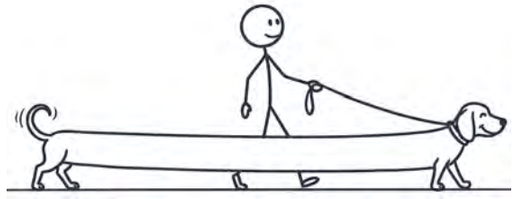
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)





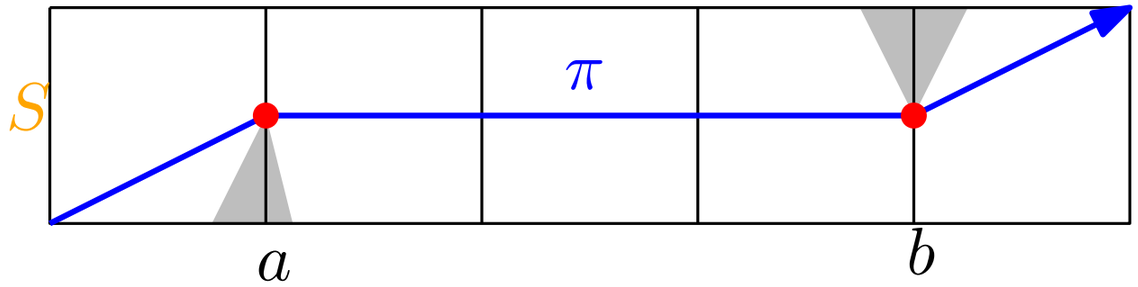
Under Transformation

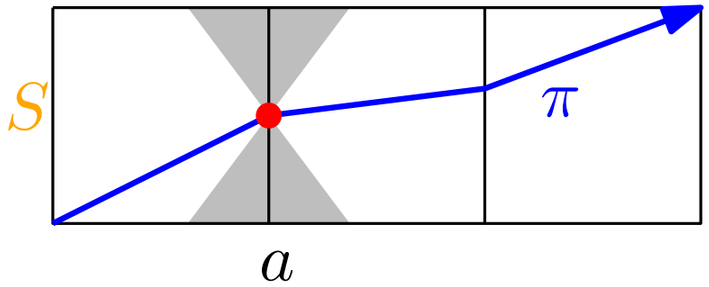
RUB

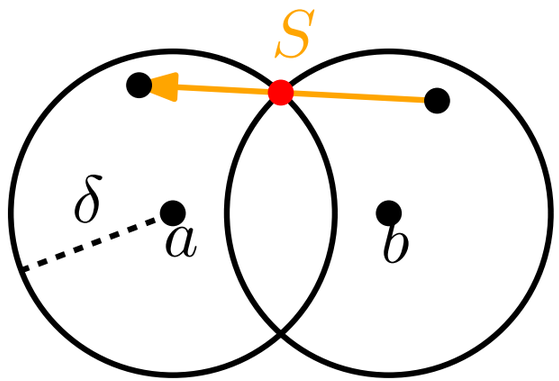
Fréchet distance of Curves under Transformations

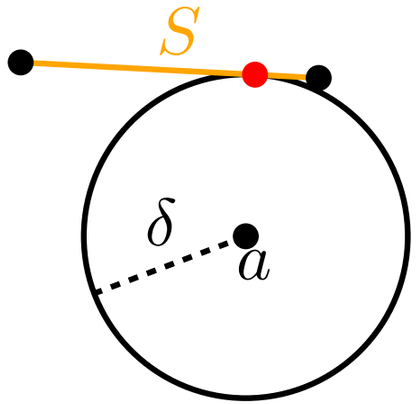
- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?









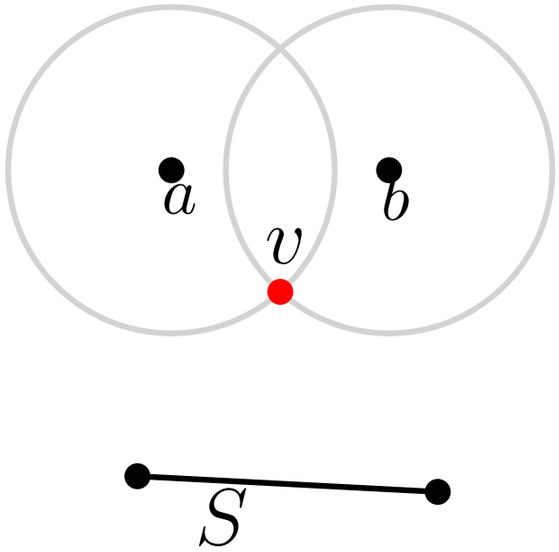
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space

Translation space

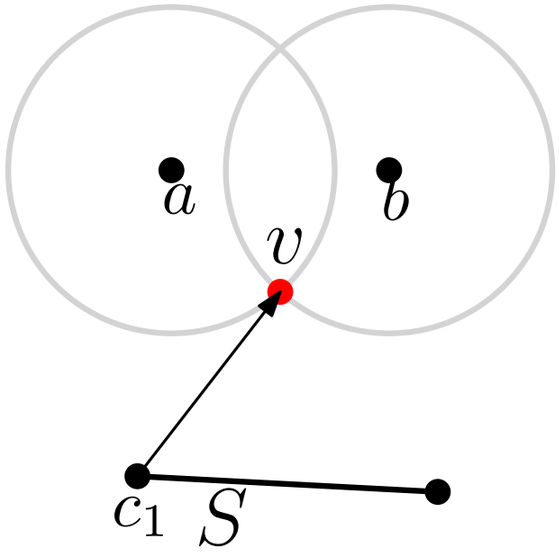
Under Transformation

RUB

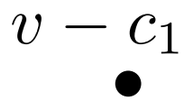
Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space



Translation space

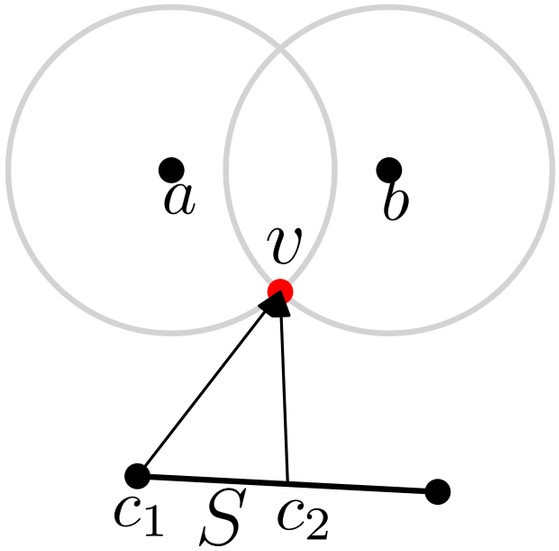
Under Transformation

RUB

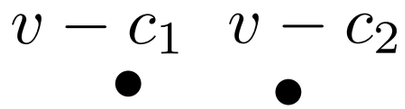
Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space



Translation space

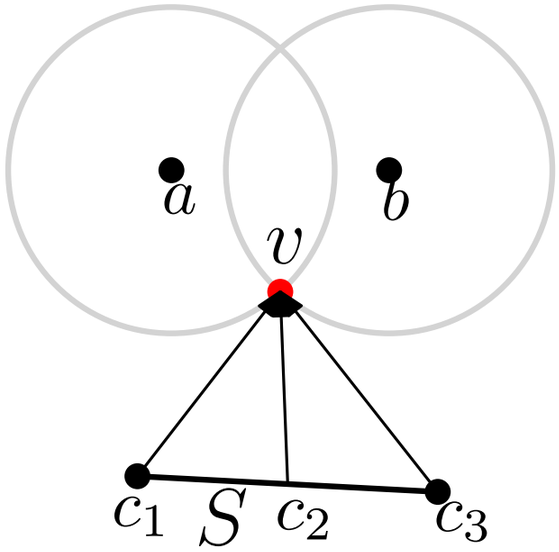
Under Transformation

RUB

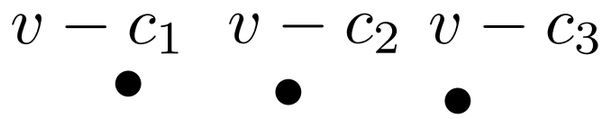
Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space



Translation space

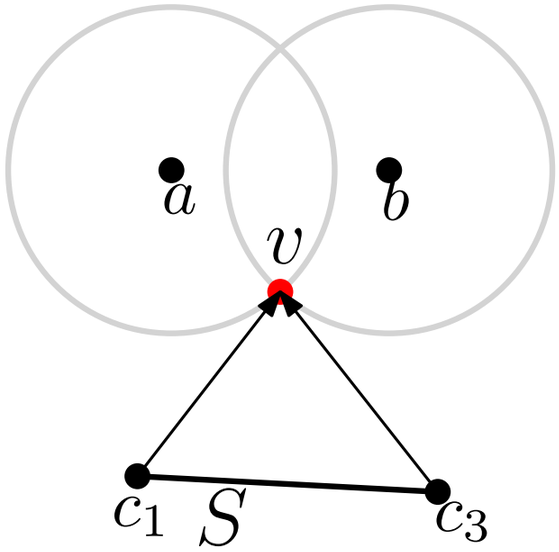
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space

- $O(n^3)$ translations in total
- $O(n^2)$ pairs of vertices on P
- $O(n)$ segments on Q

Translation space

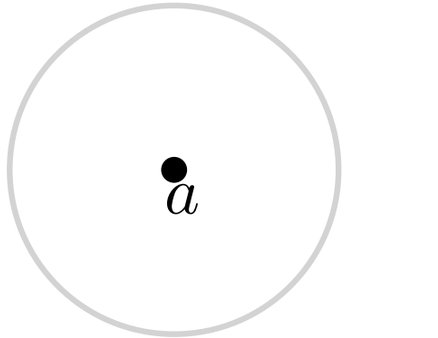
Under Transformation

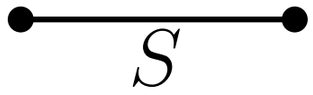
RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?





Geometric space

Translation space

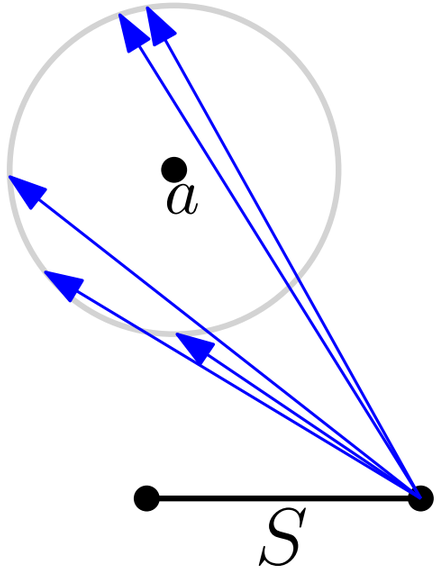
Under Transformation

RUB

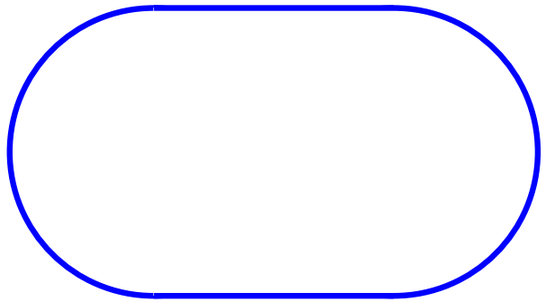
Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space



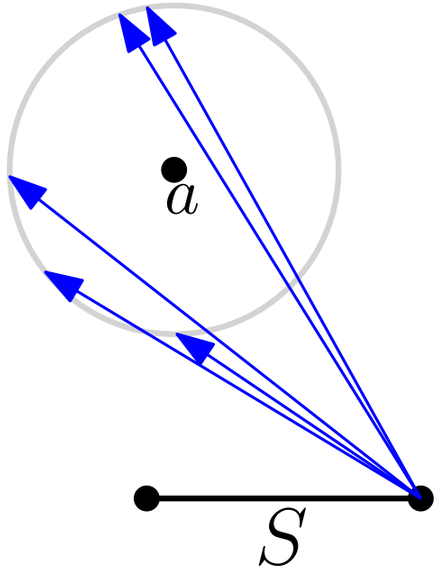
Translation space

Under Transformation

Fréchet distance of Curves under Transformations

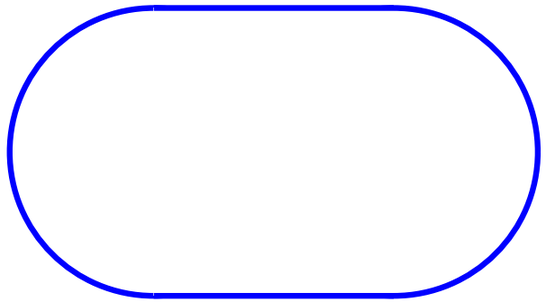
- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Geometric space

- $O(n^2)$ racetrack translations in total
- $O(n)$ vertices on Q
- $O(n)$ segments on P



Translation space

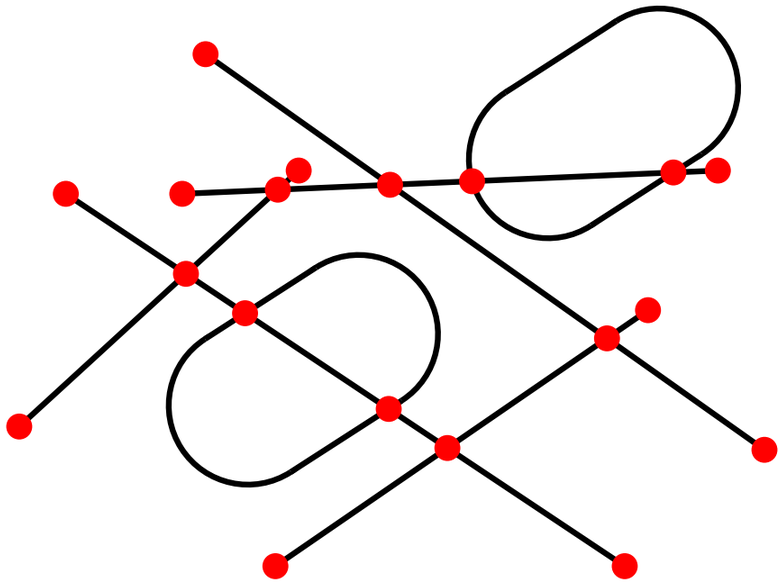
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Translation space

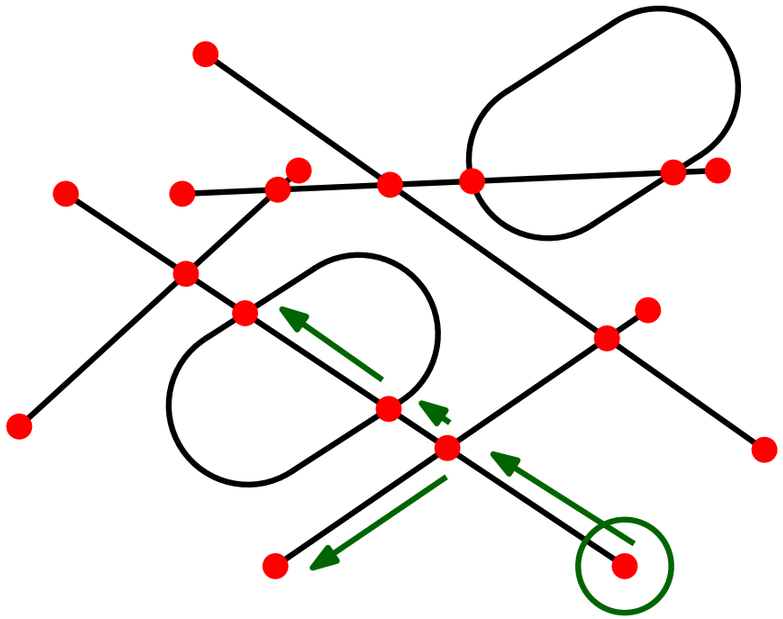
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



Translation space

1. Pick a starting intersection, and explore the arrangement of size $O(n^6)$ using DFS
2. For each intersection, update the freespace diagram and query reachability in $O(n^{\frac{4}{3}})$ time.

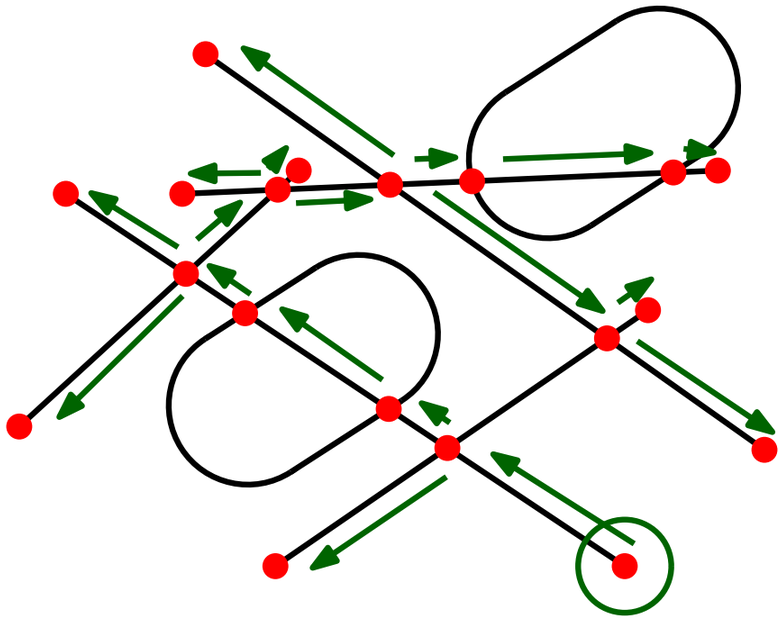
Under Transformation

RUB

Fréchet distance of Curves under Transformations

- in 2D in $\tilde{O}(n^8)$ time (Alt, Knauer, Wenk, 2001)
- in 1D in $\tilde{O}(n^{2+\frac{2}{3}})$ time (Blank et al., 2025)
- **in 2D in $\tilde{O}(n^{7+\frac{1}{3}})$ time (Buchin et al., 2025)**

which translations change reachability in fsd?



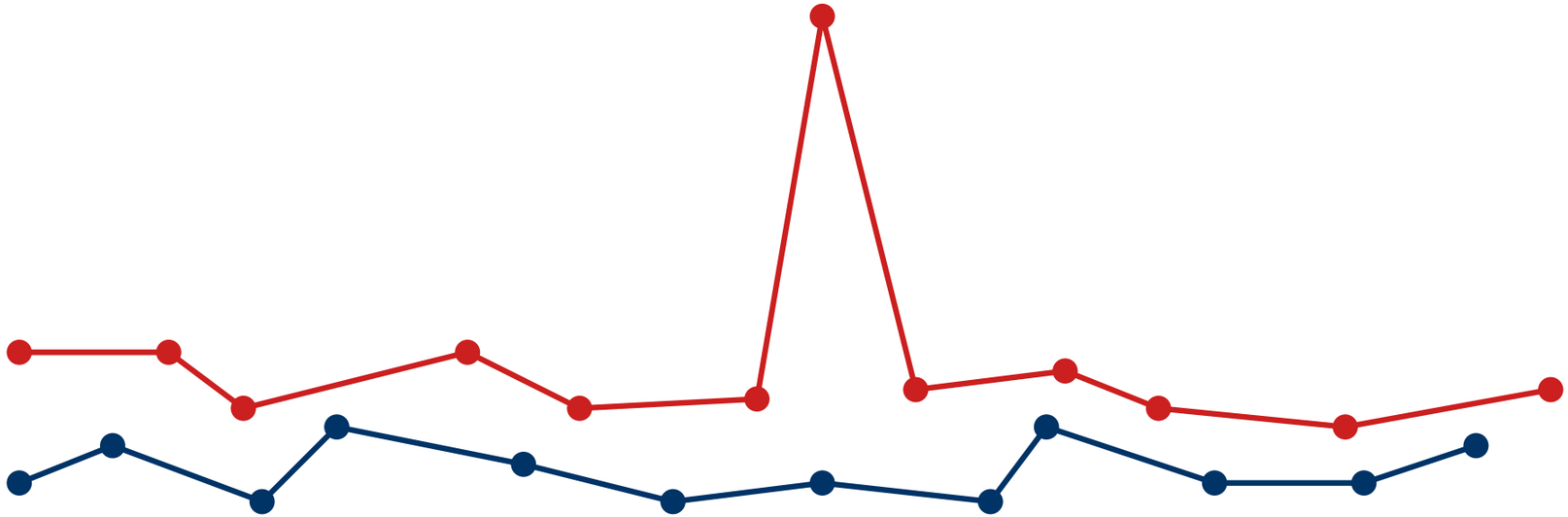
Translation space

1. Pick a starting intersection, and explore the arrangement of size $O(n^6)$ using DFS
2. For each intersection, update the freespace diagram and query reachability in $O(n^{\frac{4}{3}})$ time.

Less Sensitive to Outliers

RUB

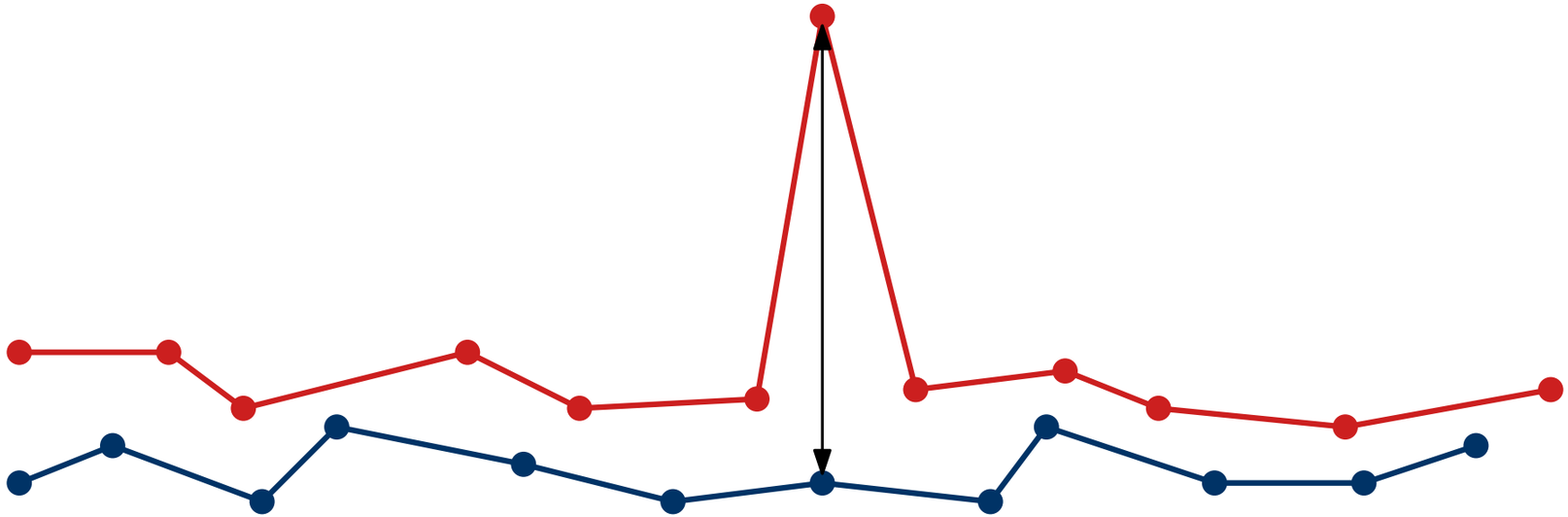
Fréchet distance is a bottleneck measure



Less Sensitive to Outliers

RUB

Fréchet distance is a bottleneck measure



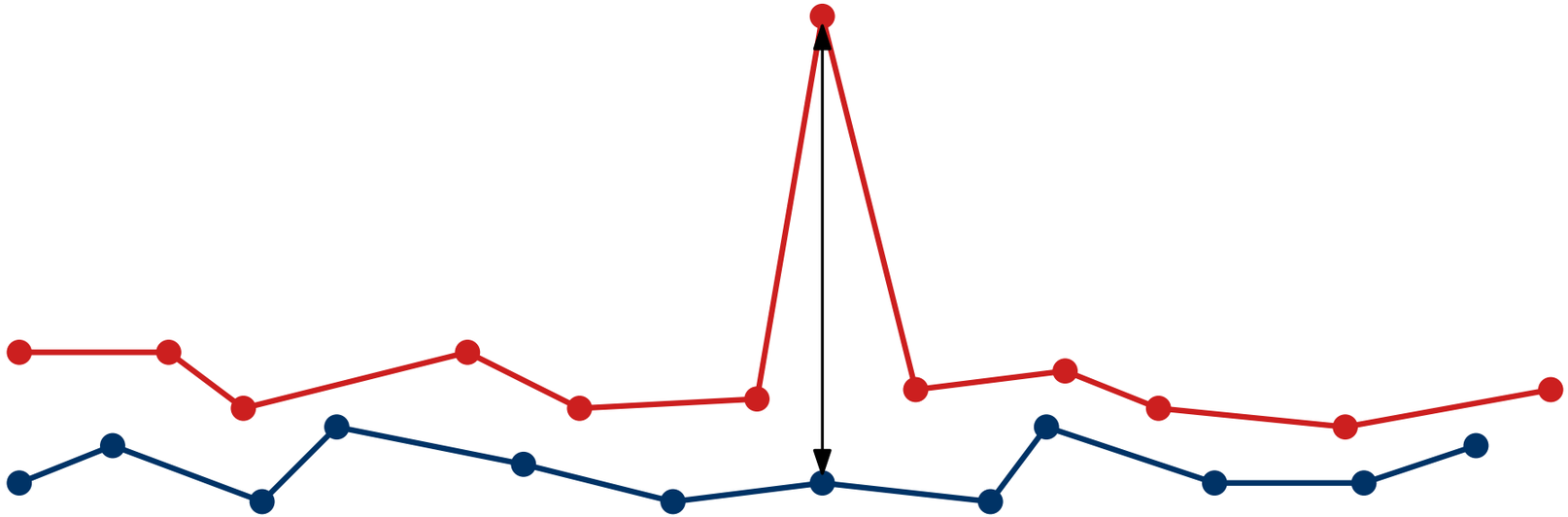
Less Sensitive to Outliers

RUB

Fréchet distance is a bottleneck measure

How to handle this?

- ignore parts of the curve
- take an average



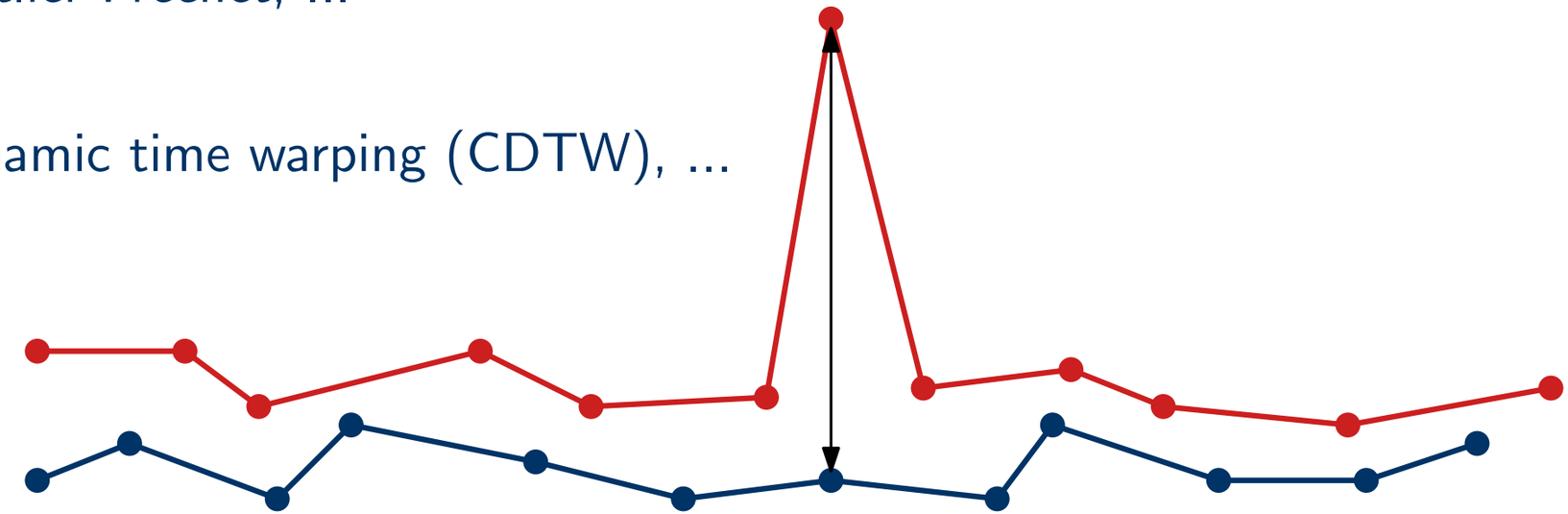
Less Sensitive to Outliers

RUB

Fréchet distance is a bottleneck measure

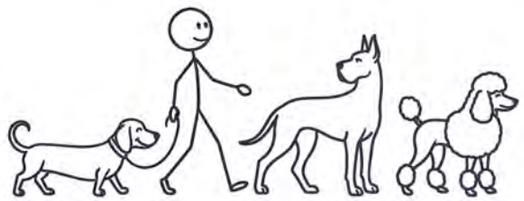
How to handle this?

- ignore parts of the curve
  partial, k-shortcut, k-outlier Fréchet, ...
- take an average
  average, continuous dynamic time warping (CDTW), ...



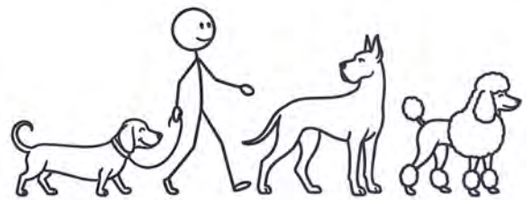
Variants

How can the Fréchet distance be adapted?



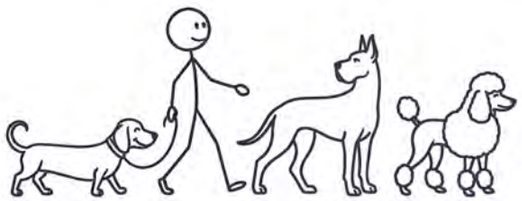
Variants

- weak [1992, Alt & Godau]



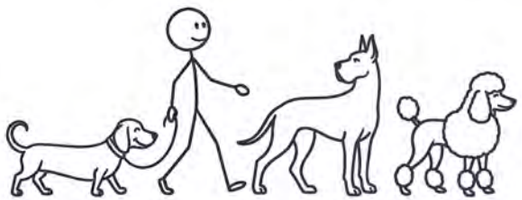
Variants

- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]



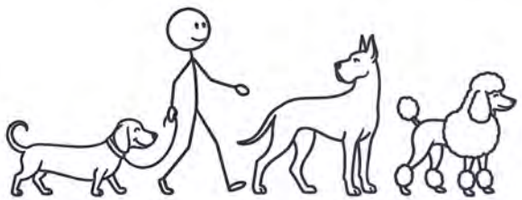
Variants

- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]



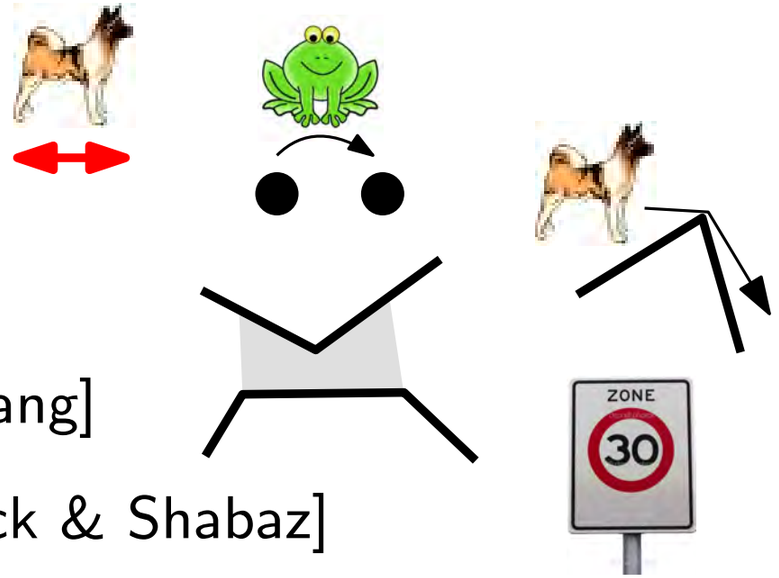
Variants

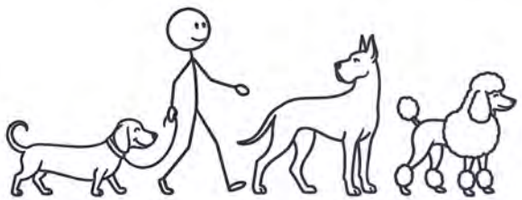
- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]



Variants

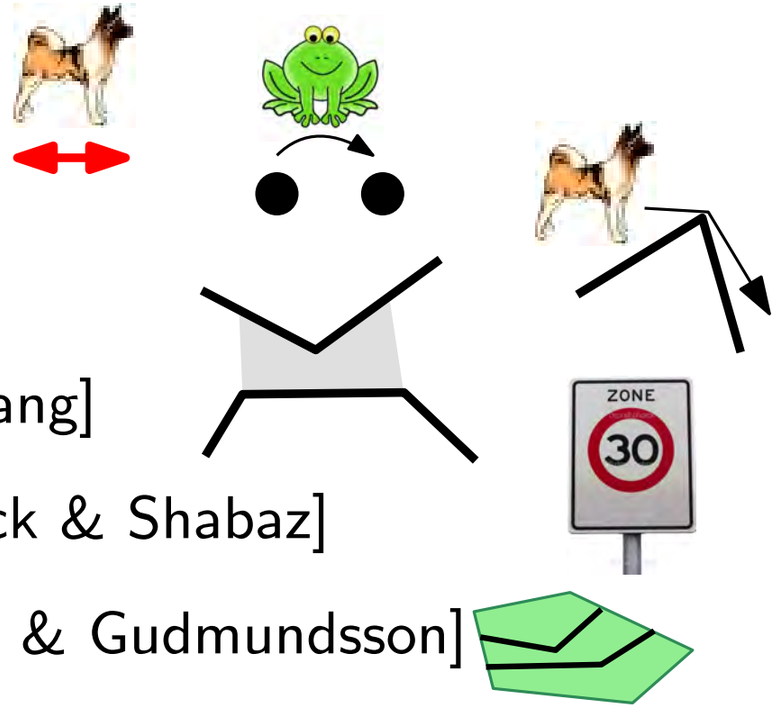
- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]

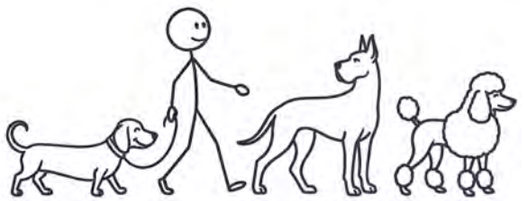




Variants

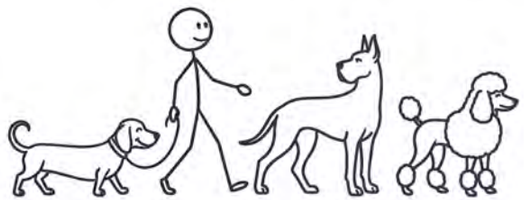
- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]
- constrained [2010, Buchin, Buchin, & Gudmundsson]





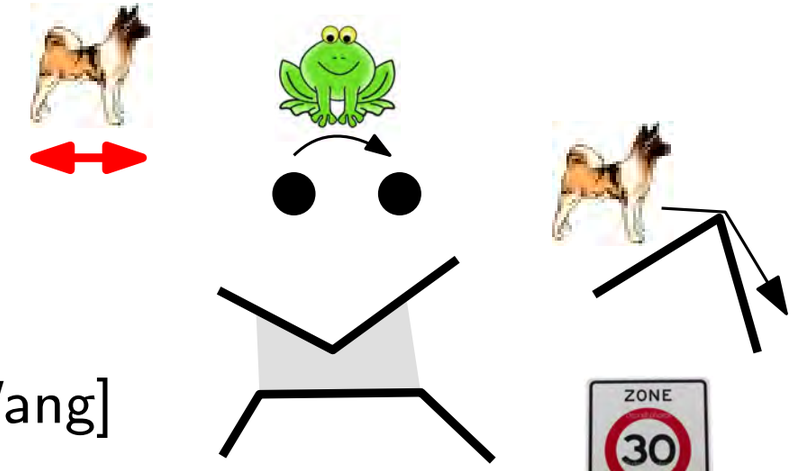
Variants

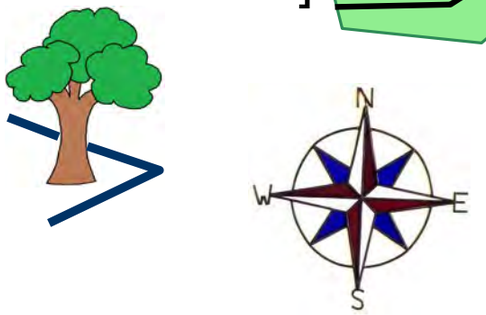
- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]
- constrained [2010, Buchin, Buchin, & Gudmundsson]
- homotopic [2010, Chambers et al.]

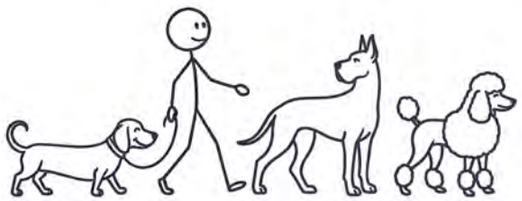


Variants

- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]
- constrained [2010, Buchin, Buchin, & Gudmundsson]
- homotopic [2010, Chambers et al.]
- direction-based [2011, de Berg & Cook]

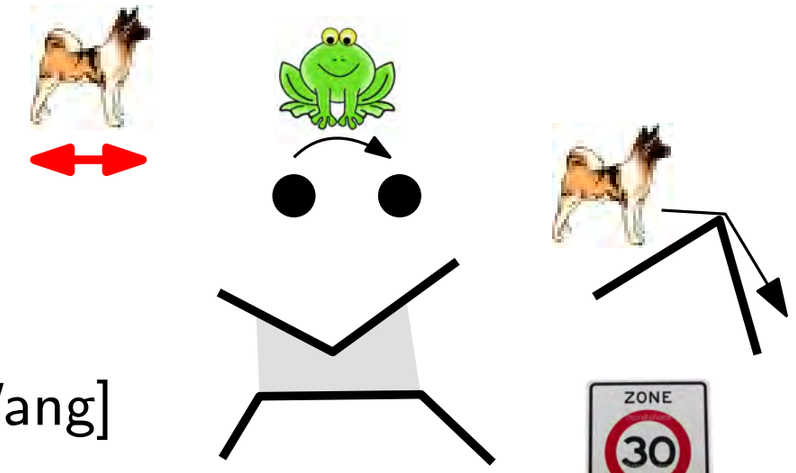


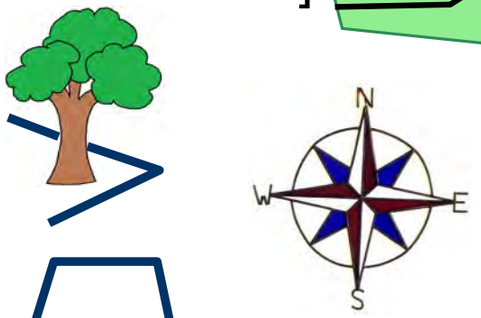


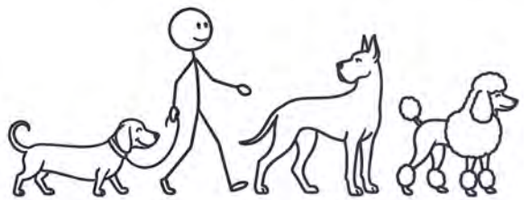


Variants

- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]
- constrained [2010, Buchin, Buchin, & Gudmundsson]
- homotopic [2010, Chambers et al.]
- direction-based [2011, de Berg & Cook]
- shortcut [2012, Driemel & Har-Peled]

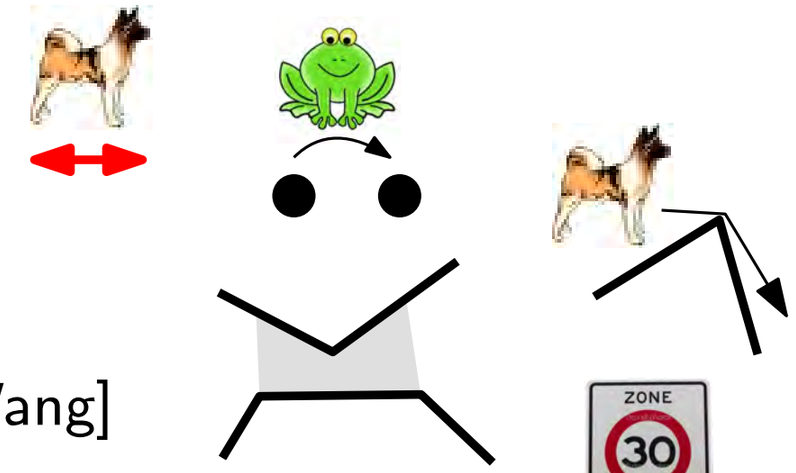


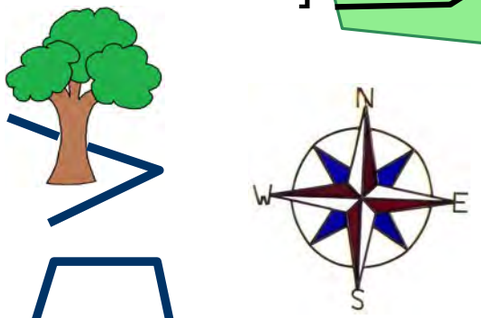


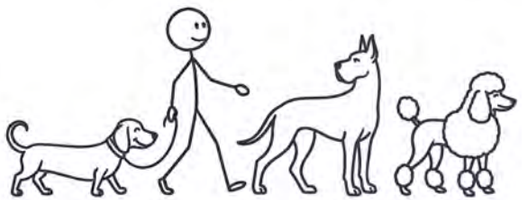


Variants

- weak [1992, Alt & Godau]
- discrete [1994, Eiter & Mannila]
- geodesic [2008, Cook & Wenk]
- partial [2009, Buchin, Buchin & Wang]
- speedlimits [2009, Maheshwari, Sack & Shabaz]
- constrained [2010, Buchin, Buchin, & Gudmundsson]
- homotopic [2010, Chambers et al.]
- direction-based [2011, de Berg & Cook]
- shortcut [2012, Driemel & Har-Peled]
- ... 14 more years of variants ..

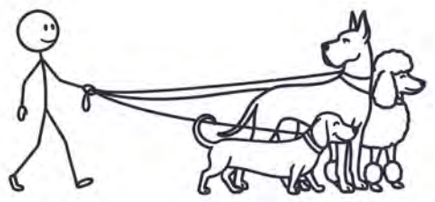






Tasks using similarity

Similarity used in many other tasks for example

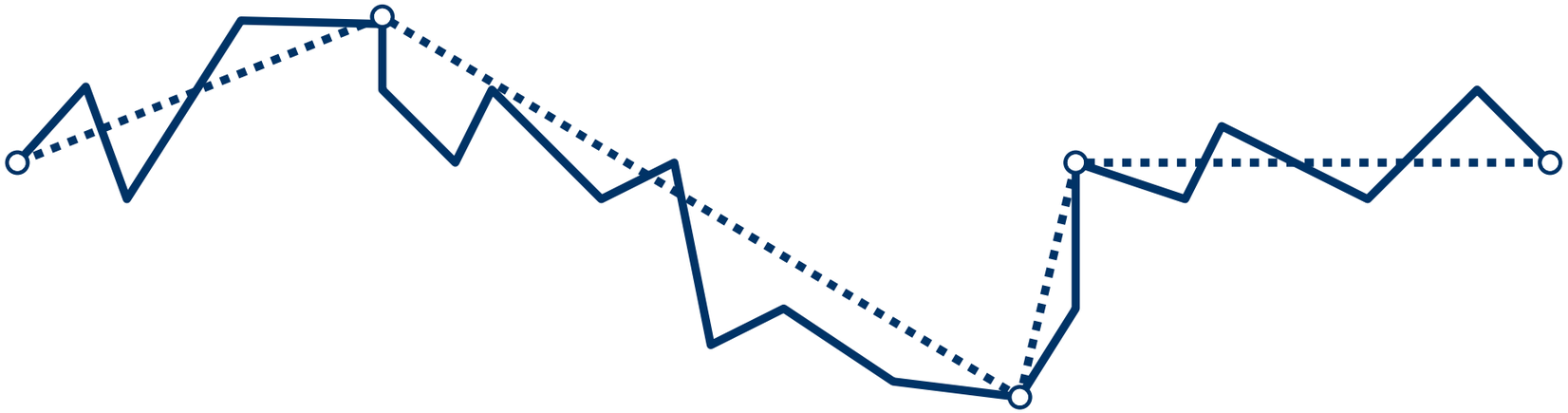


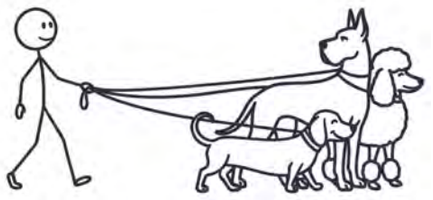
Tasks using similarity

RUB

Similarity used in many other tasks for example

- **simplification:** reduce complexity with high similarity



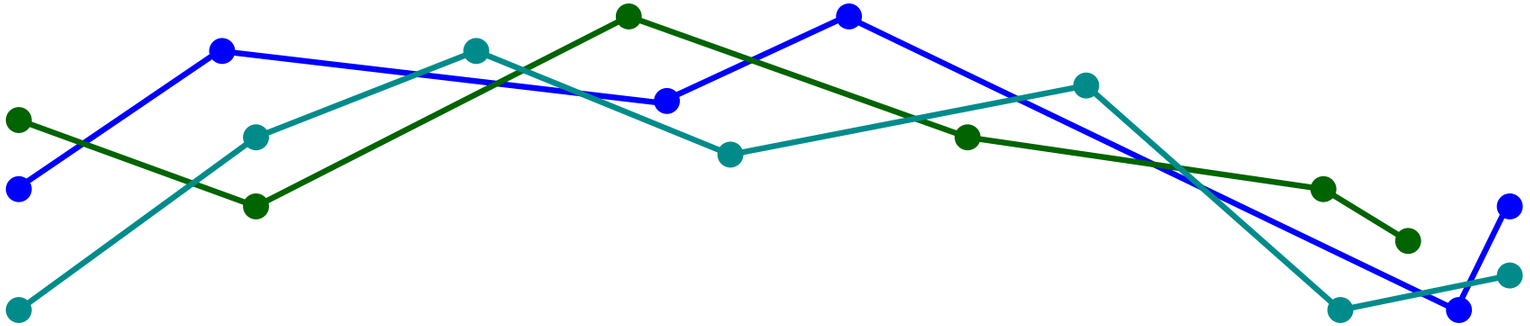


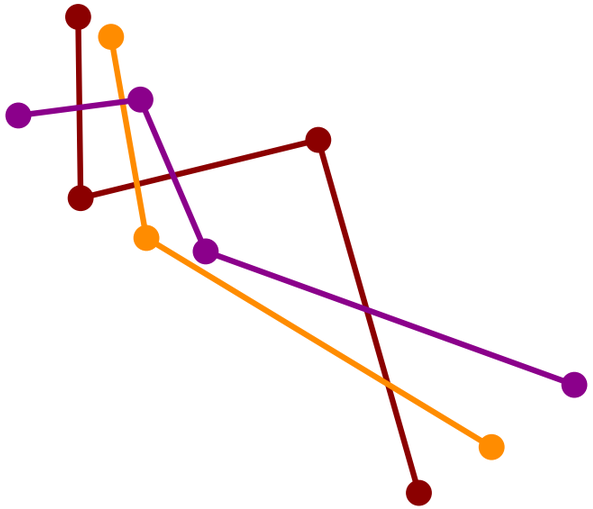
Tasks using similarity

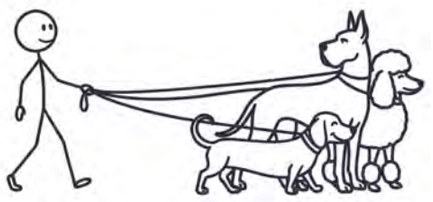
RUB

Similarity used in many other tasks for example

- **simplification:** reduce complexity with high similarity
- **clustering:** partition a set of curves into similar ones





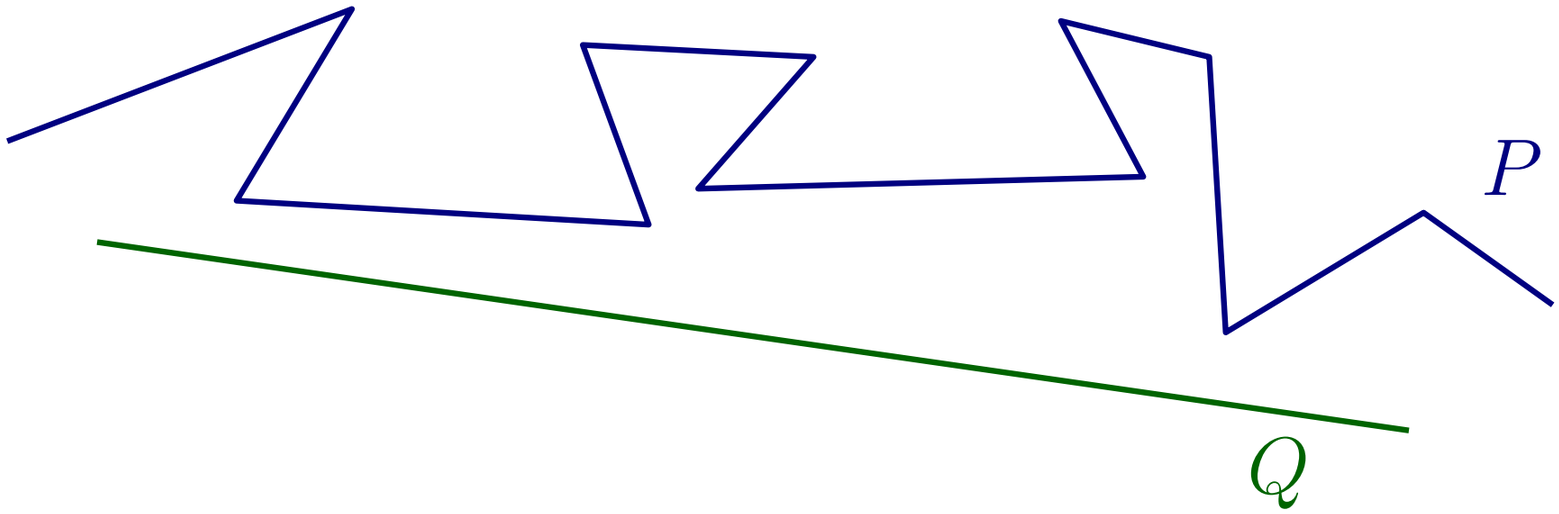


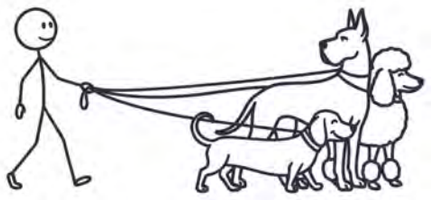
Tasks using similarity

RUB

Similarity used in many other tasks for example

- **simplification:** reduce complexity with high similarity
- **clustering:** partition a set of curves into similar ones
- **queries:** store in a data structure for faster queries



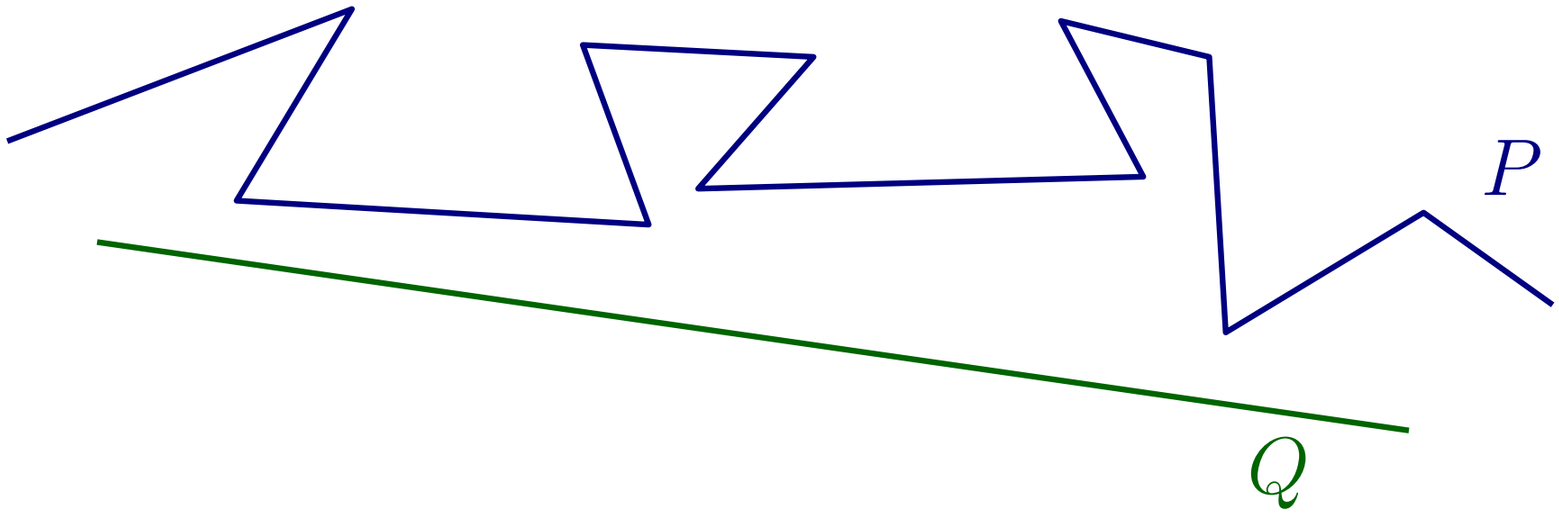


Tasks using similarity

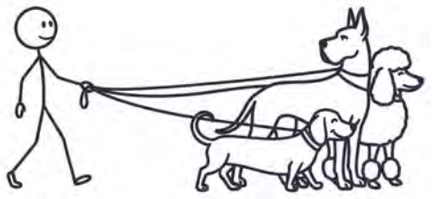
RUB

Similarity used in many other tasks for example

- **simplification:** reduce complexity with high similarity
- **clustering:** partition a set of curves into similar ones
- **queries:** store in a data structure for faster queries



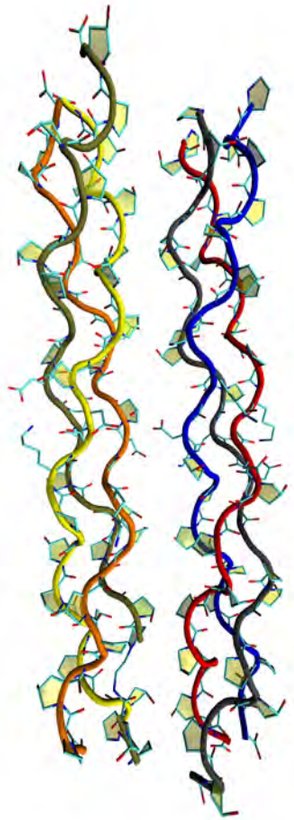
many interesting settings and results



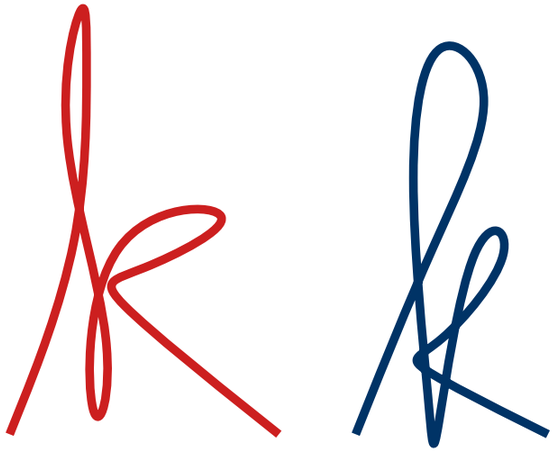
Applications

RUB

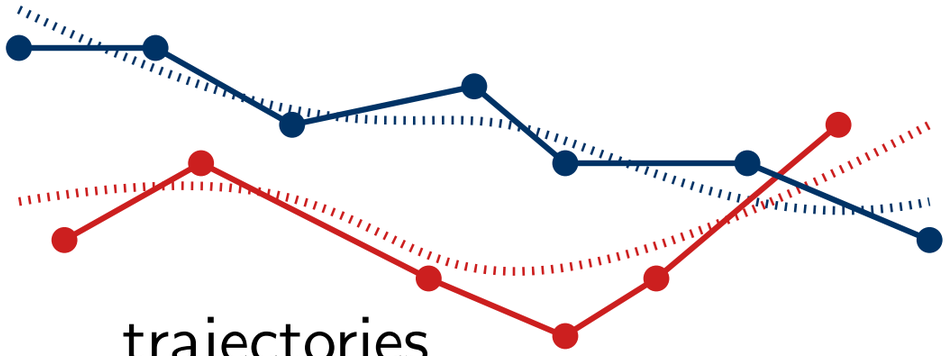
many applications, for instance..



protein backbones



handwriting



trajectories

Applications: Road Network Analysis

RUB

many entities moving
on a network

Applications: Road Network Analysis

RUB

many entities moving
on a network

Questions

- which path was taken?

Applications: Road Network Analysis

RUB

many entities moving
on a network

Questions

- which path was taken?
- which mode of transportation?

Applications: Road Network Analysis

RUB

many entities moving
on a network

Questions

- which path was taken?
- which mode of transportation?
- can we reconstruct the network?

Applications: Road Network Analysis

RUB

many entities moving
on a network

Questions

- which path was taken?
- which mode of transportation?
- can we reconstruct the network?
- how to compare two networks?

Applications: Road Network Analysis

RUB

many entities moving
on a network

Questions

- which path was taken? map matching
- which mode of transportation? segmentation, classification
- can we reconstruct the network? map construction
- how to compare two networks? map comparison

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

But also still open questions, for instance

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

But also still open questions, for instance

Approximation

- no 3-approximation in subquadratic time
- $(7 + \varepsilon)$-approximation in subquadratic time

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

But also still open questions, for instance

Approximation

- no 3-approximation in subquadratic time
- $(7 + \varepsilon)$-approximation in subquadratic time

Under Transformations

- no lower bounds in continuous case
- can we improve upper bound?

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

But also still open questions, for instance

Approximation

- no 3-approximation in subquadratic time
- $(7+\varepsilon)$-approximation in subquadratic time

Under Transformations

- no lower bounds in continuous case
- can we improve upper bound?

Generalizations

- more algorithms for surfaces
- other unsettled variants and extensions..

Summary & Open Problems

RUB

Fréchet distance has been well explored in the last 35 years!

much more research not mentioned here

But also still open questions, for instance

Approximation

- no 3-approximation in subquadratic time
- $(7+\varepsilon)$-approximation in subquadratic time

Under Transformations

- no lower bounds in continuous case
- can we improve upper bound?

Generalizations

- more algorithms for surfaces
- other unsettled variants and extensions..

Thank you!

Questions?