Approximating the Fréchet Distance for Low Highway Dimension Graphs

Anne Driemel and **Marena Richter**

University of Bonn

EuroCG '24, March 2024

The Problem

Overall problem

Given: Graph G, shortest path P, walk Q.
Goal: Compute discrete Fréchet distance of P and Q.

The Problem

Overall problem

Given: Graph G, shortest path P, walk Q.
Goal: Compute discrete Fréchet distance of P and Q.

Decision problem

Given: Graph G, shortest path P, walk Q, $\delta > 0$.
Goal: Decide if $D_{\mathcal{F}}(P, Q) \leq \delta$ or $D_{\mathcal{F}}(P, Q) > \delta$.

The Problem

Overall problem

Given: Graph G, shortest path P, walk Q.
Goal: Compute discrete Fréchet distance of P and Q.

Decision problem Approximation

Given: Graph G, shortest path P, walk Q, $\delta > 0$.
Goal: Decide if $D_{\mathcal{F}}(P, Q) \leq \alpha \cdot \delta$ or $D_{\mathcal{F}}(P, Q) > \delta$ for some $\alpha > 1$.

The Problem

Overall problem

Given: Graph G, shortest path P, walk Q.
Goal: Compute discrete Fréchet distance of P and Q.

Decision problem Approximation

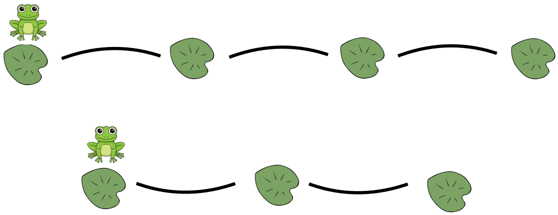
Given: Graph G, shortest path P, walk Q, $\delta > 0$.
Goal: Decide if $D_{\mathcal{F}}(P, Q) \leq \alpha \cdot \delta$ or $D_{\mathcal{F}}(P, Q) > \delta$ for some $\alpha > 1$.

Related work for decision problem:

	apx-factor	time
Driemel, van der Hoog, Rotenberg [2022]	$1 + \varepsilon$	$\mathcal{O}\left(\lvert G\rvert \log \lvert G\rvert / \sqrt{\varepsilon} + \lvert P\rvert + \frac{1}{\varepsilon}\lvert Q\rvert\right)$
van der Hoog, Rotenberg, Wong [2023]	$1 + \varepsilon$	$\mathcal{O}\left(\frac{1}{\varepsilon}\lvert Q\rvert\left(T_{dist} + \log\lvert P\rvert\right)\right)$
Driemel, Richter [2024]	$\frac{5}{3} + \varepsilon$	$\mathcal{O}\left(\lvert P\rvert + \lvert Q\rvert\left(h \log h\right)^2 \log D\right)$

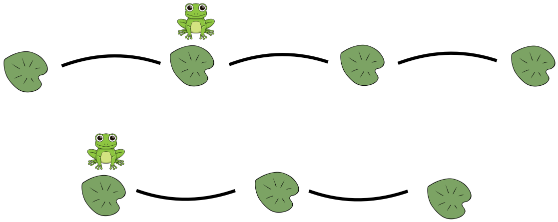
D diameter, h highway dimension

Discrete Fréchet Distance – Idea



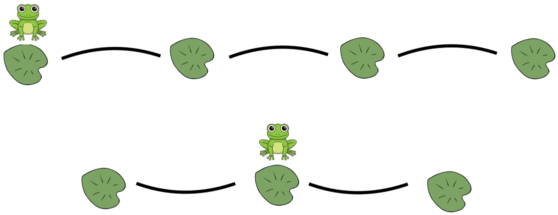
Three possible legal steps:

Discrete Fréchet Distance – Idea



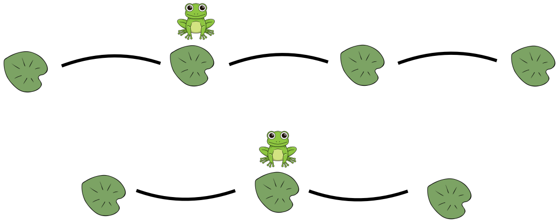
Three possible legal steps: 1) one frog jumps

Discrete Fréchet Distance – Idea



Three possible legal steps: 2) the other frog jumps

Discrete Fréchet Distance – Idea



Three possible legal steps: 3) both frogs jump

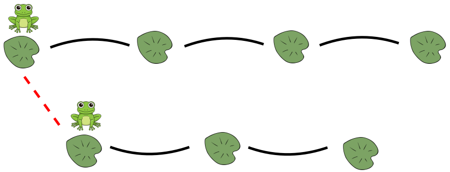
Discrete Fréchet Distance

Setting: metric graph G, distances w.r.t. shortest-path metric.

Definition (Discrete Fréchet Distance)

The **discrete Fréchet distance** of two walks $P = \langle p_1, p_2, \ldots, p_n \rangle$ and $Q = \langle q_1, q_2, \ldots, q_m \rangle$ in G is the minimum over the maximum pairwise distance of any legal traversal $\mathcal{T} \in P \times Q$:

$$D_{\mathcal{F}}(P, Q) := \min_{\mathcal{T}} \max_{(p_i, p_j) \in \mathcal{T}} \text{dist}_G(p_i, q_i).$$



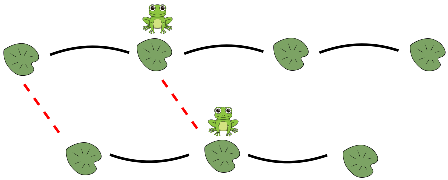
Discrete Fréchet Distance

Setting: metric graph G, distances w.r.t. shortest-path metric.

Definition (Discrete Fréchet Distance)

The **discrete Fréchet distance** of two walks $P = \langle p_1, p_2, \ldots, p_n \rangle$ and $Q = \langle q_1, q_2, \ldots, q_m \rangle$ in G is the minimum over the maximum pairwise distance of any legal traversal $\mathcal{T} \in P \times Q$:

$$D_{\mathcal{F}}(P, Q) := \min_{\mathcal{T}} \max_{(p_i, p_j) \in \mathcal{T}} \mathrm{dist}_G(p_i, q_i).$$



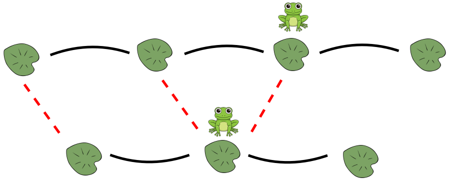
Discrete Fréchet Distance

Setting: metric graph G, distances w.r.t. shortest-path metric.

Definition (Discrete Fréchet Distance)

The **discrete Fréchet distance** of two walks $P = \langle p_1, p_2, \ldots, p_n \rangle$ and $Q = \langle q_1, q_2, \ldots, q_m \rangle$ in G is the minimum over the maximum pairwise distance of any legal traversal $\mathcal{T} \in P \times Q$:

$$D_{\mathcal{F}}(P, Q) := \min_{\mathcal{T}} \max_{(p_i, p_j) \in \mathcal{T}} \mathrm{dist}_G(p_i, q_i).$$



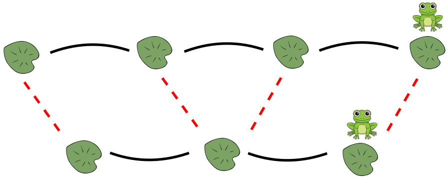
Discrete Fréchet Distance

Setting: metric graph G, distances w.r.t. shortest-path metric.

Definition (Discrete Fréchet Distance)

The **discrete Fréchet distance** of two walks $P = \langle p_1, p_2, \ldots, p_n \rangle$ and $Q = \langle q_1, q_2, \ldots, q_m \rangle$ in G is the minimum over the maximum pairwise distance of any legal traversal $\mathcal{T} \in P \times Q$:

$$D_{\mathcal{F}}(P, Q) := \min_{\mathcal{T}} \max_{(p_i, p_j) \in \mathcal{T}} \text{dist}_G(p_i, q_i).$$



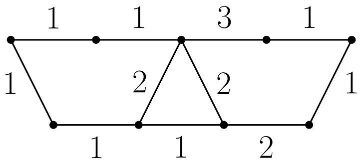
Free Space Matrix

Definition (Free-Space Matrix)

Given two walks P and Q in G, $\delta > 0$, the **free-space matrix** $M_\delta \in \mathbb{R}^{|P|\times|Q|}$ is defined by

$$M_\delta[i,j] = \begin{cases} 1 & \text{dist}_G(p_i, q_j) \leq \delta \\ 0 & \text{else} \end{cases}$$

G :



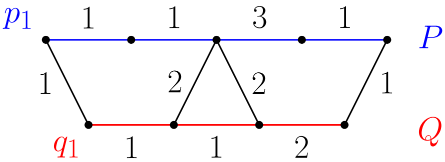
Free Space Matrix

Definition (Free-Space Matrix)

Given two walks P and Q in G, $\delta > 0$, the **free-space matrix** $M_\delta \in \mathbb{R}^{|P| \times |Q|}$ is defined by

$$M_\delta[i,j] = \begin{cases} 1 & \mathrm{dist}_G(p_i, q_j) \leq \delta \\ 0 & \text{else} \end{cases}$$

G :



M_2 :

	0	0	0	1	1
	0	0	1	0	0
	1	0	1	0	0
q_1	1	1	0	0	0
	p_1				

Free Space Matrix

Definition (Free-Space Matrix)

Given two walks P and Q in G, $\delta > 0$, the **free-space matrix** $M_\delta \in \mathbb{R}^{|P| \times |Q|}$ is defined by

$$M_\delta[i,j] = \begin{cases} 1 & \text{dist}_G(p_i, q_j) \leq \delta \\ 0 & \text{else} \end{cases}$$

Observation:

$D_\mathcal{F}(P,Q) \leq \delta$

$\Leftrightarrow$ $\exists$ legal traversal $\mathcal{T}$ through $P \times Q$ s.t. $M_\delta[i,j] = 1$ $\forall (p_i, q_j) \in \mathcal{T}$.

$$\begin{array}{cccccc} M_2: & 0 & 0 & 0 & 1 & 1 \\ & 0 & 0 & 1 & 0 & 0 \\ & 1 & 0 & 1 & 0 & 0 \\ q_1 & 1 & 1 & 0 & 0 & 0 \\ & p_1 & & & & \end{array}$$

Free Space Matrix

Definition (Free-Space Matrix)

Given two walks P and Q in G, $\delta > 0$, the **free-space matrix** $M_\delta \in \mathbb{R}^{|P| \times |Q|}$ is defined by

$$M_\delta[i,j] = \begin{cases} 1 & \text{dist}_G(p_i, q_j) \leq \delta \\ 0 & \text{else} \end{cases}$$

Observation:

$$D_{\mathcal{F}}(P,Q) \leq \delta$$
$$\Leftrightarrow \exists \text{ legal traversal } \mathcal{T} \text{ through } P \times Q \text{ s.t. } M_\delta[i,j] = 1$$
$$\forall (p_i, q_j) \in \mathcal{T}.$$

M_2 :

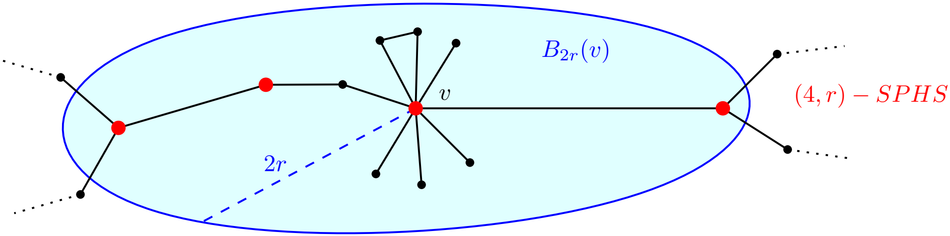
	0	0	0	1	1
	0	0	1	0	0
	1	0	1	0	0
q_1	1	1	0	0	0
	p_1				

Sparse Shortest Path Hitting Sets

Definition (Sparse Shortest Path Hitting Set (SPHS))

For $r > 0$ an (h, r)-**SPHS** is a set $C \subseteq V(G)$ s.t.

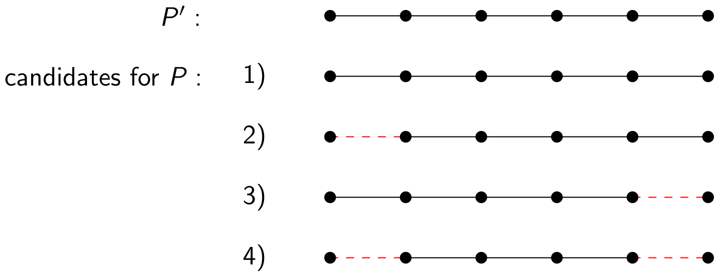
- $|B_{2r}(v) \cap C| \leq h$ for all $v \in V(G)$,
- $V(P) \cap C \neq \emptyset$ for all "long (w.r.t. r)" shortest paths P.



r-significant paths

Definition (r-significant path)

A shortest path P' is an r-**witness** for P if $\ell(P') > r$ and $P = P'$ or P arises from P' by deleting one or both end vertices. P is r-**significant** if it has an r-witness.



Sparse Shortest Path Hitting Sets

Definition (Sparse Shortest Path Hitting Set (SPHS))

For $r > 0$ an (h, r)-**SPHS** is a set $C \subseteq V(G)$ s.t.

- $|B_{2r}(v) \cap C| \leq h$ for all $v \in V(G)$,
- $V(P) \cap C \neq \emptyset$ for all r-significant shortest paths P.

Sparse Shortest Path Hitting Sets

Definition (Sparse Shortest Path Hitting Set (SPHS))

For $r > 0$ an (h, r)-**SPHS** is a set $C \subseteq V(G)$ s.t.

- $|B_{2r}(v) \cap C| \leq h$ for all $v \in V(G)$,
- $V(P) \cap C \neq \emptyset$ for all r-significant shortest paths P.

Idea: Highway dimension $\approx$ smallest h s.t. (h, r)-SPHS exists for all $r > 0$.

Sparse Shortest Path Hitting Sets

Definition (Sparse Shortest Path Hitting Set (SPHS))

For $r > 0$ an (h, r)-**SPHS** is a set $C \subseteq V(G)$ s.t.

- $|B_{2r}(v) \cap C| \leq h$ for all $v \in V(G)$,
- $V(P) \cap C \neq \emptyset$ for all r-significant shortest paths P.

Idea: Highway dimension $\approx$ smallest h s.t. (h, r)-SPHS exists for all $r > 0$.

Theorem (Abraham, Delling, Fiat, Goldberg, Werneck)

In a graph with highway dimension h, we can compute a $(\mathcal{O}(h \log h), r)$-SPHS in polynomial runtime.

The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$
Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \delta$ or $D_{\mathcal{F}}(P, Q) > \delta$.

The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$
Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$
Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\to C$.

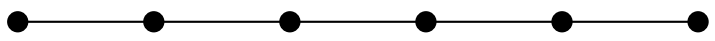
The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\to C$.
2. compute simplification of $P \to P^{\delta}$:

P :



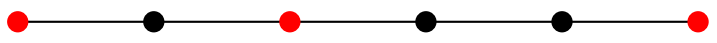
The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$.
2. compute simplification of $P \rightarrow P^\delta$:
   visits endpoints of P and $V(P) \cap C$.

P^δ, P :

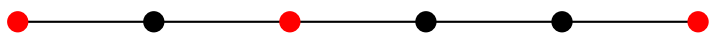


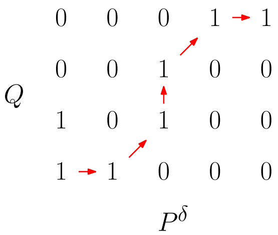
The Algorithm

Given: shortest path P, arbitrary walk Q, value $\delta > 0$
Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$.
2. compute simplification of $P \rightarrow P^\delta$:
   visits endpoints of P and $V(P) \cap C$.
3. BFS through $M_{2\delta}$ of P^δ and $Q \rightarrow D_{\mathcal{F}}\left(P^\delta, Q\right) \leq 2\delta$?

P^δ, P :





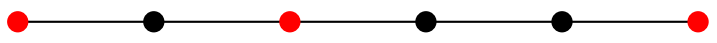
The Algorithm

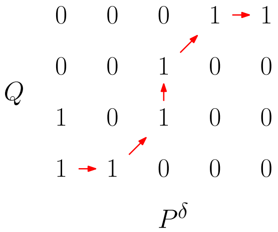
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$.
2. compute simplification of $P \rightarrow P^{\delta}$: visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}\left(P^{\delta}, P\right) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^{δ} and $Q \rightarrow D_{\mathcal{F}}\left(P^{\delta}, Q\right) \leq 2\delta$?

P^{δ}, P :





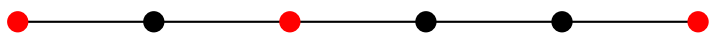
The Algorithm

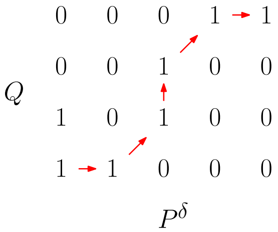
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$.
2. compute simplification of $P \rightarrow P^\delta$:
   visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}(P^\delta, P) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^δ and $Q \rightarrow D_{\mathcal{F}}(P^\delta, Q) \leq 2\delta$?
   Δ-inequality: $\Rightarrow D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$

P^δ, P :





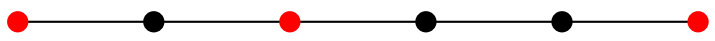
The Algorithm

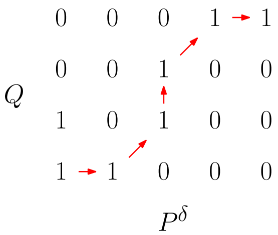
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$. $\rightarrow$ polynomial
2. compute simplification of $P \rightarrow P^\delta$: visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}(P^\delta, P) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^δ and $Q \rightarrow D_{\mathcal{F}}(P^\delta, Q) \leq 2\delta$?
   Δ-inequality: $\Rightarrow D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$

P^δ, P :





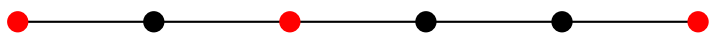
The Algorithm

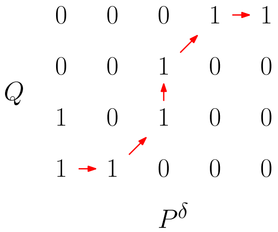
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\to C$. $\to$ polynomial
2. compute simplification of $P \to P^{\delta}$: $\to \mathcal{O}(|P|)$
   visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}(P^{\delta}, P) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^{δ} and $Q \to D_{\mathcal{F}}(P^{\delta}, Q) \leq 2\delta$?
   Δ-inequality: $\Rightarrow D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$

P^{δ}, P :





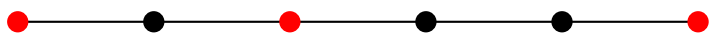
The Algorithm

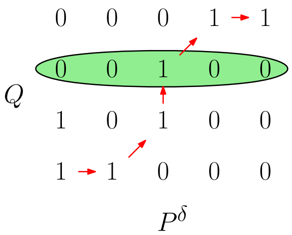
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P,Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P,Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$. $\rightarrow$ polynomial
2. compute simplification of $P \rightarrow P^\delta$: $\rightarrow \mathcal{O}(|P|)$
   visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}(P^\delta, P) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^δ and $Q \rightarrow D_{\mathcal{F}}(P^\delta, Q) \leq 2\delta$?
   Δ-inequality: $\Rightarrow D_{\mathcal{F}}(P,Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P,Q) > \frac{3}{2}\delta$
   $\#$(non-zero entries in each row) $\leq h' + 2$

P^δ, P :





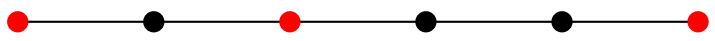
The Algorithm

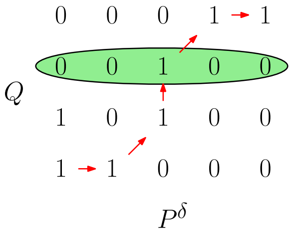
Given: shortest path P, arbitrary walk Q, value $\delta > 0$

Goal: Decide whether $D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$.

1. compute (h', δ)-SPHS $\rightarrow C$. $\rightarrow$ polynomial
2. compute simplification of $P \rightarrow P^{\delta}$: $\rightarrow \mathcal{O}(|P|)$
   visits endpoints of P and $V(P) \cap C$. $\Rightarrow D_{\mathcal{F}}(P^{\delta}, P) \leq \frac{\delta}{2}$
3. BFS through $M_{2\delta}$ of P^{δ} and $Q \rightarrow D_{\mathcal{F}}(P^{\delta}, Q) \leq 2\delta$?
   Δ-inequality: $\Rightarrow D_{\mathcal{F}}(P, Q) \leq \frac{5}{2}\delta$ or $D_{\mathcal{F}}(P, Q) > \frac{3}{2}\delta$
   $\#$(non-zero entries in each row) $\leq h' + 2$
   $\Rightarrow \mathcal{O}(|Q|\, h')$ steps

P^{δ}, P :





The Algorithm

Problems:

1. Computing C can take long.

The Algorithm

Problems:

1. Computing C can take long.
2. Need to compute distances in G during the BFS.

The Algorithm

Problems:

1. Computing C can take long.
2. Need to compute distances in G during the BFS.

Solutions:

The Algorithm

Problems:

1. Computing C can take long.
2. Need to compute distances in G during the BFS.

Solutions:

1. pre-compute ε-**multiscale SPHS** for $\varepsilon > 1$: (h', ε^{i-1})-SPHS C_i for $0 \leq i \leq \lceil \log D / \log \varepsilon \rceil$.

$D :=$ diameter of G.

The Algorithm

Problems:

1. Computing C can take long.
2. Need to compute distances in G during the BFS.

Solutions:

1. pre-compute ε-**multiscale SPHS** for $\varepsilon > 1$: (h', ε^{i-1})-SPHS C_i for $0 \leq i \leq \lceil \log D / \log \varepsilon \rceil$.
   In the algorithm: Choose C_i such that $\varepsilon^{i-1} \approx \delta$.

$D :=$ diameter of G.

The Algorithm

Problems:

1. Computing C can take long.
2. Need to compute distances in G during the BFS.

Solutions:

1. pre-compute ε-**multiscale SPHS** for $\varepsilon > 1$: (h', ε^{i-1})-SPHS C_i for $0 \leq i \leq \lceil \log D / \log \varepsilon \rceil$.
   In the algorithm: Choose C_i such that $\varepsilon^{i-1} \approx \delta$.
2. Use oracle with $\mathcal{O}(h' \log D)$ query time using a 2-multiscale SPHS.

$D :=$ diameter of G.

Final Result

Theorem

G metric graph with highway dimension h, $\varepsilon > 0$.

Preprocessing G in time polynomial in $|V(G)|$ and $1/\log(1+\varepsilon)$ using $\mathcal{O}(|V(G)| \log D(1/\log(1+\varepsilon) + h \log h))$ space.

$\Rightarrow$ decide for any shortest path P, walk Q, $\delta > 0$, if $D_{\mathcal{F}}(P, Q) > \delta$ or $D_{\mathcal{F}}(P, Q) \leq (\frac{5}{3} + \varepsilon)\delta$ in $\mathcal{O}\left(|P| + |Q| (h \log h)^2 \log D\right)$ time.

Final Result

Theorem

G metric graph with highway dimension h, $\varepsilon > 0$.

Preprocessing G in time polynomial in $|V(G)|$ and $1/\log(1+\varepsilon)$ using $\mathcal{O}(|V(G)| \log D(1/\log(1+\varepsilon) + h \log h))$ space.

$\Rightarrow$ decide for any shortest path P, walk Q, $\delta > 0$, if $D_{\mathcal{F}}(P, Q) > \delta$ or $D_{\mathcal{F}}(P, Q) \leq (\frac{5}{3} + \varepsilon)\delta$ in $\mathcal{O}\left(|P| + |Q| (h \log h)^2 \log D\right)$ time.

Remark

Binary search $\Rightarrow$ $(\frac{5}{3} + \varepsilon)$-approximation of $D_{\mathcal{F}}(P, Q)$.

Final Result

	apx-factor	time
Driemel, van der Hoog, Rotenberg [2022]	$1+\varepsilon$	$\mathcal{O}\left(\lvert G\rvert \log \lvert G\rvert / \sqrt{\varepsilon} + \lvert P\rvert + \frac{1}{\varepsilon}\lvert Q\rvert\right)$
van der Hoog, Rotenberg, Wong [2023]	$1+\varepsilon$	$\mathcal{O}\left(\frac{1}{\varepsilon}\lvert Q\rvert \left(T_{dist} + \log \lvert P\rvert\right)\right)$
Driemel, Richter [2024]	$\frac{5}{3}+\varepsilon$	$\mathcal{O}\left(\lvert P\rvert + \lvert Q\rvert (h \log h)^2 \log D\right)$

Thank you for your attention!

Questions?