Fast Distance Queries

Laurent Viennot

MPRI - Graph Mining 2/8
Problem
Given a graph $G$ assign a label $L_u$ to each node $u$ s.t. for all $s, t \ d(s, t)$ can be computed from $L_s$ and $L_t$.

Hub sets
Given a graph $G$, assign a hub set $H_u \subseteq V(G)$ to each node $u$, s.t. for all $u, v$ there exists $a \in H_u \cap H_v$ with $a \in P_{uv}$.

Distance labels: $L_u = \{(a, d(u, a)) : a \in H_u\}$

Distance query: $\text{Dist} (L_s, L_t) = \min_{a \in H_s \cap H_t} d(s, a) + d(a, t)$ in $O(|H_s| + |H_t|)$ time.
Distance labeling [Gavoille, Peleg, Pérennes, Raz ’04]

Problem
Given a graph $G$ assign a label $L_u$ to each node $u$ s.t. for all $s, t$ $d(s, t)$ can be computed from $L_s$ and $L_t$.

Hub sets
Given a graph $G$, assign a hub set $H_u$ $\subseteq V(G)$ to each node $u$, s.t. for all $u, v$ there exists $a \in H_u \cap H_v$ with $a \in P_{uv}$.

Distance labels: $L_u = \{(a, d(u, a)) : a \in H_u\}$

Distance query: $\text{Dist} (L_s, L_t) = \min_{a \in H_s \cap H_t} d(s, a) + d(a, t)$ in $O(|H_s| + |H_t|)$ time.
Distance labeling [Gavoille, Peleg, Pérennes, Raz ’04]

Problem
Given a graph G assign a label $L_u$ to each node $u$ s.t. for all $s, t$ $d(s, t)$ can be computed from $L_s$ and $L_t$.

Hub sets
Given a graph G, assign a hub set $H_u \subseteq V(G)$ to each node $u$, s.t. for all $u, v$ there exists $a \in H_u \cap H_v$ with $a \in P_{uv}$.

Distance labels : $L_u = \{(a, d(u, a)) : a \in H_u\}$

Distance query : $\text{Dist} (L_s, L_t) = \min_{a \in H_s \cap H_t} d(s, a) + d(a, t)$
in $O(|H_s| + |H_t|)$ time.
Distance labeling [Gavoille, Peleg, Pérennes, Raz ’04]

Problem
Given a graph $G$ assign a label $L_u$ to each node $u$ s.t. for all $s, t$ $d(s, t)$ can be computed from $L_s$ and $L_t$.

Hub sets
Given a graph $G$, assign a hub set $H_u \subseteq V(G)$ to each node $u$, s.t. for all $u, v$ there exists $a \in H_u \cap H_v$ with $a \in P_{uv}$.

Distance labels: $L_u = \{(a, d(u, a)) : a \in H_u\}$

Distance query: $\text{Dist} (L_s, L_t) = \min_{a \in H_s \cap H_t} d(s, a) + d(a, t)$
in $O(|H_s| + |H_t|)$ time.
Distance labeling [Gavoille, Peleg, Pérennes, Raz ’04]

Problem
Given a graph $G$ assign a label $L_u$ to each node $u$ s.t. for all $s, t$ $d(s, t)$ can be computed from $L_s$ and $L_t$.

Hub sets
Given a graph $G$, assign a hub set $H_u \subseteq V(G)$ to each node $u$, s.t. for all $u, v$ there exists $a \in H_u \cap H_v$ with $a \in P_{uv}$.

Distance labels : $L_u = \{(a, d(u, a)) : a \in H_u\}$

Distance query : $\text{Dist} (L_s, L_t) = \min_{a \in H_s \cap H_t} d(s, a) + d(a, t)$ in $O(|H_s| + |H_t|)$ time.
Labeling with hub sets: hub labeling a.k.a. 2-hop labeling

Exercise: Hub labeling for a path?, a tree? a graph with treewidth k? a planar graph?

Open pb: Increase the best known lower bound for unweighted planar graphs ($\Omega(n^{1/3})$ [Gavoille et al. 2004]).

Exercise: Hub labeling for a grid?
Labeling with hub sets: hub labeling a.k.a. 2-hop labeling

Exercise: Hub labeling for a path?, a tree? a graph with treewidth k? a planar graph?

Open pb: Increase the best known lower bound for unweighted planar graphs ($\Omega(n^{1/3})$ [Gavoille et al. 2004]).

Exercise: Hub labeling for a grid?
Labeling with hub sets: hub labeling a.k.a. 2-hop labeling

**Exercise**: Hub labeling for a path?, a tree? a graph with treewidth k? a planar graph?

**Open pb**: Increase the best known lower bound for unweighted planar graphs ($\Omega(n^{1/3})$ [Gavoille et al. 2004]).

**Exercise**: Hub labeling for a grid?
2-hop labeling [Cohen, Halperin, Kaplan, Zwick ’03]

**Greedy cover** all shortest paths:
- smallest avg. hub size is equivalent to min. cost set cover,
- $O(\log n)$-approximation is possible:
  - select $a, S$ s.t. $\frac{\text{nb path cov. if } a \text{ added to all } (H_u)_{u \in S}}{|S|} \max$.

**Problem**: set cover instance with $n \times 2^n$ sets!

**Solution**: fix $a$, what is the best $S$?
- $G_a$ graph with edges $uv$ s.t. $P_{uv}$ uncov. and $a \in P_{uv}$.

**Exercise**: Propose a greedy algorithm for 2-approximating the best $S$. Hint: average degree $\delta$ increases when removing a node with degree $< \delta/2$.

**Corollary**: Hubsets with smallest average size can be $O(\log n)$-approximated in polynomial time.
2-hop labeling [Cohen, Halperin, Kaplan, Zwick ’03]

**Greedy cover** all shortest paths:
- smallest avg. hub size is equivalent to min. cost set cover,
- $O(\log n)$-approximation is possible:
  - select $a, S$ s.t. \[
  \text{nb path cov. if } a \text{ added to all } (H_u)_{u \in S} \max \frac{|S|}{|S|} \]

**Problem**: set cover instance with $n \times 2^n$ sets!

**Solution**: fix $a$, what is the best $S$?

$G_a$ graph with edges $uv$ s.t. $P_{uv}$ uncov. and $a \in P_{uv}$.

**Exercise**: Propose a greedy algorithm for 2-approximating the best $S$. Hint: average degree $\delta$ increases when removing a node with degree $< \delta/2$.

**Corollary**: Hubsets with smallest average size can be $O(\log n)$-approximated in polynomial time.
2-hop labeling [Cohen, Halperin, Kaplan, Zwick ’03]

**Greedy cover** all shortest paths:
- smallest avg. hub size is equivalent to min. cost set cover,
- $O(\log n)$-approximation is possible:
  - select $a, S$ s.t. \[
  \text{nb path cov. if } a \text{ added to all } (H_u)_{u \in S} \max
  \]

**Problem**: set cover instance with $n \times 2^n$ sets!

**Solution**: fix $a$, what is the best $S$?
- $G_a$ graph with edges $uv$ s.t. $P_{uv}$ uncov. and $a \in P_{uv}$.

**Exercise**: Propose a greedy algorithm for 2-approximating the best $S$. Hint: average degree $\delta$ increases when removing a node with degree $< \delta/2$.

**Corollary**: Hubsets with smallest average size can be $O(\log n)$-approximated in polynomial time.
2-hop labeling [Cohen, Halperin, Kaplan, Zwick ‘03]

**Greedy cover** all shortest paths:
- smallest avg. hub size is equivalent to min. cost set cover,
- $O(\log n)$-approximation is possible:
- select $a, S$ s.t. $\frac{\text{nb path cov. if } a \text{ added to all } (H_u)_{u \in S}}{|S|}$ max.

**Problem**: set cover instance with $n \times 2^n$ sets!

**Solution**: fix $a$, what is the best $S$?
- $G_a$ graph with edges $uv$ s.t. $P_{uv}$ uncov. and $a \in P_{uv}$.

**Exercise**: Propose a greedy algorithm for 2-approximating the best $S$. Hint: average degree $\delta$ increases when removing a node with degree $< \delta/2$.

**Corollary**: Hubsets with smallest average size can be $O(\log n)$-approximated in polynomial time.
2-hop labeling [Cohen, Halperin, Kaplan, Zwick ’03]

**Greedy cover** all shortest paths:
- smallest avg. hub size is equivalent to min. cost set cover,
- $O(\log n)$-approximation is possible:
  - select $a, S$ s.t. $\frac{\text{nb path cov. if } a \text{ added to all } (H_u)_{u \in S}}{|S|}$ max.

**Problem**: set cover instance with $n \times 2^n$ sets!

**Solution**: fix $a$, what is the best $S$?
- $G_a$ graph with edges $uv$ s.t. $P_{uv}$ uncov. and $a \in P_{uv}$.

**Exercise**: Propose a greedy algorithm for 2-approximating the best $S$. Hint: average degree $\delta$ increases when removing a node with degree $< \delta/2$.

**Corollary**: Hubsets with smallest average size can be $O(\log n)$-approximated in polynomial time.
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

**Graph property** ensuring small hub sets and efficient ordering for CH.

**Definition**
Highway dimension $h = \max_{u,r} \min_H \text{hitting set of } \mathcal{P}_{ur} |H|$
where $\mathcal{P}_{ur} = \{P \in \mathcal{P}_r | \bar{P} \cap B(u, r) \neq \emptyset\}$, $\mathcal{P}_r = \{P | \ell(P) > \frac{r}{2}\}$, and $\bar{P}$ is any shortest path extending $P$ by 0 or 1 edge at each extremity.

**Theorem**
Any graph $G$ with highway dimension $h$ and diameter $D$ has hub sets of size $O(h \log D)$ ($O(h \log h \log D)$ for polyn. time).

**Lemma**
For all $r$, $G$ has an $(h, r)$-sparse hitting set, i.e. a set $C$ s.t. $C \cap P \neq \emptyset$ for all $P \in \mathcal{P}_r$ and $|B(u, r) \cap C| \leq h$ for all $u \in V(G)$. 
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

**Graph property** ensuring small hub sets and efficient ordering for CH.

**Definition**

Highway dimension \( h = \max_{u, r} \min_{H} |H| \) hitting set of \( \mathcal{P}_{ur} \)

where \( \mathcal{P}_{ur} = \{ P \in \mathcal{P}_r | \overline{P} \cap B(u, r) \neq \emptyset \} \), \( \mathcal{P}_r = \{ P | \ell(P) > \frac{r}{2} \} \),

and \( \overline{P} \) is any shortest path extending \( P \) by 0 or 1 edge at each extremity.

**Theorem**

Any graph \( G \) with highway dimension \( h \) and diameter \( D \) has hub sets of size \( O(h \log D) \) \( (O(h \log h \log D) \) for polyn. time).

**Lemma**

For all \( r, G \) has an \( (h, r) \)-sparse hitting set, i.e. a set \( C \) s.t. \( C \cap P \neq \emptyset \) for all \( P \in \mathcal{P}_r \) and \( |B(u, r) \cap C| \leq h \) for all \( u \in V(G) \).
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Graph property ensuring small hub sets and efficient ordering for CH.

Definition
Highway dimension $h = \max_{u, r} \min_H |H|$ hitting set of $P_{ur}$
where $P_{ur} = \{P \in \mathcal{P}_r | \bar{P} \cap B(u, r) \neq \emptyset\}$, $\mathcal{P}_r = \{P | \ell(P) > \frac{r^2}{2}\}$, and $\bar{P}$ is any shortest path extending $P$ by 0 or 1 edge at each extremity.

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ has hub sets of size $O(h \log D)$ ($O(h \log h \log D)$ for polyn. time).

Lemma
For all $r$, $G$ has an $(h, r)$-sparse hitting set, i.e. a set $C$ s.t. $C \cap P \neq \emptyset$ for all $P \in \mathcal{P}_r$ and $|B(u, r) \cap C| \leq h$ for all $u \in V(G)$. 

⇒ ? ⇒
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Graph property ensuring small hub sets and efficient ordering for CH.

Definition
Highway dimension $h = \max_{u,r} \min_{H} \text{hitting set of } P_{u,r} |H|
where $P_{u,r} = \{P \in P_r : \bar{P} \cap B(u,r) \neq \emptyset\}$, $P_r = \{P : \ell(P) > \frac{r}{2}\}$,
and $\bar{P}$ is any shortest path extending $P$ by 0 or 1 edge at each extremity.

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ has
hub sets of size $O(h \log D)$ ($O(h \log h \log D)$ for polyn. time).

Lemma
For all $r$, $G$ has an $(h, r)$-sparse hitting set, i.e. a set $C$ s.t.
$C \cap P \neq \emptyset$ for all $P \in P_r$ and $|B(u, r) \cap C| \leq h$ for all $u \in V(G)$. 
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH^\text{opt}_\pi$ produces at most $O(nh \log D)$ shortcuts and $CH_\pi + RP$ bidir. Dij. visits $O(h \log D)$ nodes.

Hierarchical Hub Labeling (HHL) [BGKSW’15]
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

Canonical HHL
Given an ordering $\pi$, for all $u, v$ add $\max_{\pi} P_{uv}$ to $H_u$ and $H_v$.

Proposition
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

Exercise: show that any minimal HHL is canonical.

Exercise: use CH to compute a HHL.
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH_{\pi}^{opt}$ produces at most $O(hn \log D)$ shortcuts and $CH_{\pi} + RP$ bidir. Dij. visits $O(h \log D)$ nodes.

Hierarchical Hub Labeling (HHL) [BGKSW’15]
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

Canonical HHL
Given an ordering $\pi$, for all $u, v$ add $\max_{\pi} P_{uv}$ to $H_u$ and $H_v$.

Proposition
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

Exercise: show that any minimal HHL is canonical.
Exercise: use CH to compute a HHL.
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

**Theorem**
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH_{\pi}^{opt}$ produces at most $O(\text{nh log } D)$ shortcuts and $CH_\pi + RP$ bidir. Dij. visits $O(h \log D)$ nodes.

**Hierarchical Hub Labeling (HHL) [BGKSW’15]**
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

**Canonical HHL**
Given an ordering $\pi$, for all $u, v$ add $\max_\pi p_{uv}$ to $H_u$ and $H_v$.

**Proposition**
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

**Exercise**: show that any minimal HHL is canonical.
**Exercise**: use CH to compute a HHL.
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH^{\text{opt}}_{\pi}$ produces at most $O(nh \log D)$ shortcuts and $CH_{\pi} + \text{RP bidir. Dij. visits}$ $O(h \log D)$ nodes.

Hierarchical Hub Labeling (HHL) [BGKSW’15]
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

Canonical HHL
Given an ordering $\pi$, for all $u, v$ add $\max_{\pi} P_{uv}$ to $H_u$ and $H_v$.

Proposition
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

Exercise: show that any minimal HHL is canonical.
Exercise: use CH to compute a HHL.
Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH_{\pi}^{opt}$ produces at most $O(nh \log D)$ shortcuts and $CH_{\pi} + RP$ bidir. Dij. visits $O(h \log D)$ nodes.

Hierarchical Hub Labeling (HHL) [BGKSW’15]
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

Canonical HHL
Given an ordering $\pi$, for all $u, v$ add $\max_{\pi} P_{uv}$ to $H_u$ and $H_v$.

Proposition
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

Exercise: show that any minimal HHL is canonical.
Exercise: use CH to compute a HHL.
Highway dimension [Abraham, Delling, Fiat, Goldberg, Werneck ’10-13]

Theorem
Any graph $G$ with highway dimension $h$ and diameter $D$ admits a node ordering $\pi$ s.t. $CH^\text{opt}_\pi$ produces at most $O(nh \log D)$ shortcuts and $CH_\pi + RP$ bidir. Dij. visits $O(h \log D)$ nodes.

Hierarchical Hub Labeling (HHL) [BGKSW’15]
A hub labeling is hierarchical if it respects an order $\pi$ such that hubs are more important: $v \in H_u \Rightarrow \pi(u) \leq \pi(v)$ (the graph with edges from nodes to their hubs is a DAG).

Canonical HHL
Given an ordering $\pi$, for all $u, v$ add $\max_{\pi} P_{uv}$ to $H_u$ and $H_v$.

Proposition
Canonical HHL for $\pi$ is the minimum HHL that respects $\pi$.

Exercise: show that any minimal HHL is canonical.
Exercise: use CH to compute a HHL.
Pruned Labeling [Akiba, Iwata, Yoshida ’13]

Procedure **PrunedLab** \((G, \pi)\)

- Distance labels \(L_u := \emptyset\) for all \(u\).
- For each \(a \in V(G)\) in decreasing order of \(\pi\) do
  - **PrunedDijkstra** \((G, a, L)\)
  - Add \((a, d(a, u))\) to \(L_u\) for each visited node \(u\).

Procedure **PrunedDijkstra** \((G, a, L)\)

- Starting from \(u = a\), visit \(u\) if \(d(u) < \text{Dist}(L_a, L_u)\).

**Theorem**

PL computes the canonical HHL associated to \(\pi\) in \(O(nL \log n + mL^2)\) time where \(L\) is maximum label size.

**Exercise** : \(O(\log n)\) approximation for HHL (find a good \(\pi\)).

**Open pb** : charac. classes of graphs with \(|HHL| = O(|HL|)\).
Pruned Labeling [Akiba, Iwata, Yoshida ’13]

**Procedure** PrunedLab \((G, \pi)\)

- Distance labels \(L_u := \emptyset\) for all \(u\).
- For each \(a \in V(G)\) in decreasing order of \(\pi\) do
  - PrunedDijkstra \((G, a, L)\)
  - Add \((a, d(a, u))\) to \(L_u\) for each visited node \(u\).

**Procedure** PrunedDijkstra \((G, a, L)\)

- Starting from \(u = a\), visit \(u\) if \(d(u) < \text{Dist}(L_a, L_u)\).

**Theorem**

PL computes the canonical HHL associated to \(\pi\) in \(O(nL \log n + mL^2)\) time where \(L\) is maximum label size.

**Exercise** : \(O(\log n)\) approximation for HHL (find a good \(\pi\)).

**Open pb** : charac. classes of graphs with \(|\text{HHL}| = O(|\text{HL}|)\).
Pruned Labeling [Akiba, Iwata, Yoshida '13]

Procedure **PrunedLab** \((G, \pi)\)

- Distance labels \(L_u := \emptyset\) for all \(u\).
- For each \(a \in V(G)\) in decreasing order of \(\pi\) do
  - **PrunedDijkstra** \((G, a, L)\)
  - Add \((a, d(a, u))\) to \(L_u\) for each visited node \(u\).

Procedure **PrunedDijkstra** \((G, a, L)\)

- Starting from \(u = a\), visit \(u\) if \(d(u) < \text{Dist}(L_a, L_u)\).

**Theorem**

PL computes the canonical HHL associated to \(\pi\) in \(O(nL \log n + mL^2)\) time where \(L\) is maximum label size.

**Exercise** : \(O(\log n)\) approximation for HHL (find a good \(\pi\)).

**Open pb** : charac. classes of graphs with \(|\text{HHL}| = O(|\text{HL}|)\).
HL on massive networks [Delling, Goldberg, Pajor, Werneck ’14]

HHL using random sampling to approximate greedy cover (for $\pi$) in combination with pruned labeling (for hub sets).

$O(\log n)$ approximation in theory, smallest hub labelings in practice.
HL on massive networks [Delling, Goldberg, Pajor, Werneck ‘14]

HHL using random sampling to approximate greedy cover (for $\pi$) in combination with pruned labeling (for hub sets).

$O(\log n)$ approximation in theory, smallest hub labelings in practice.
Skeleton dimension [Kosowski, V. ’17]

Graph property ensuring small hub sets.

The skeleton dimension $k$ of $G$ is the maximum “width” of a “pruned” shortest path tree (see pres.).

**Theorem**
Any graph $G$ with skeleton dimension $k$ and diameter $D$ has hub sets of size $O(k \log \log k \log D)$ (polyn. time constr. w.h.p.).

**Open pb**: what additional property ensures efficient Reach/CH?

**Open pb**: tight bounds on HL/HHL in grids?
Skeleton dimension [Kosowski, V. ’17]

Graph property ensuring small hub sets.

The skeleton dimension $k$ of $G$ is the maximum “width” of a “pruned” shortest path tree (see pres.).

**Theorem**
Any graph $G$ with skeleton dimension $k$ and diameter $D$ has hub sets of size $O(k \log \log k \log D)$ (polyn. time constr. w.h.p.).

**Open pb**: what additional property ensures efficient Reach/CH?

**Open pb**: tight bounds on HL/HHL in grids?
Graph property ensuring small hub sets.

The *skeleton dimension* $k$ of $G$ is the maximum “width” of a “pruned” shortest path tree (see pres.).

**Theorem**
Any graph $G$ with skeleton dimension $k$ and diameter $D$ has hub sets of size $O(k \log \log k \log D)$ (polyn. time constr. w.h.p.).

**Open pb**: what additional property ensures efficient Reach/CH?

**Open pb**: tight bounds on HL/HHL in grids?
Graph property ensuring small hub sets.

The skeleton dimension $k$ of $G$ is the maximum "width" of a "pruned" shortest path tree (see pres.).

**Theorem**

Any graph $G$ with skeleton dimension $k$ and diameter $D$ has hub sets of size $O(k \log \log k \log D)$ (polyn. time constr. w.h.p.).

**Open pb**: what additional property ensures efficient Reach/CH?

**Open pb**: tight bounds on HL/HHL in grids?
Skeleton dimension [Kosowski, V. ’17]

Graph property ensuring small hub sets.

The skeleton dimension \( k \) of \( G \) is the maximum “width” of a “pruned” shortest path tree (see pres.).

**Theorem**
Any graph \( G \) with skeleton dimension \( k \) and diameter \( D \) has hub sets of size \( O(k \log \log k \log D) \) (polyn. time constr. w.h.p.).

**Open pb**: what additional property ensures efficient Reach/CH?

**Open pb**: tight bounds on HL/HHL in grids?
Barcelona shortest path tree
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Barcelona skeleton: prune last third
Tree skeleton
Tree skeleton

\[ P_{uv}^{2/3} \]

\[ \left\{ w \in P_{uv} : d(u,w) \leq \frac{2}{3} d(u,v) \right\} \]
Tree skeleton

\[ \text{Reach}_{P_{uv}}(w) \geq \frac{d(u, w)}{2} \]
Tree skeleton

\[ T_u = \bigcup_{v \in \mathcal{P}_{uv}} \]
Tree skeleton

\[ T_u^* = U_P^{2/3} \]

\[ T_u = U_P \]
Tree skeleton

\[ T_u^* = \bigcup_{v} \mathcal{P}_{uv}^{2/3} \]

\[ T_u = \bigcup_{v} \mathcal{P}_{uv} \]

\( \text{Cut}_u(T_u) \)
Tree skeleton

Width($T_u^*$) = $\max_{r \in R} |\text{Cut}_r(T_u^*)|$
Tree skeleton

$$\text{skel. dim. } k = \max_u \text{Width}(T_u^*)$$

$$T_u^* = \bigvee_{\nu} P_{\nu u}^{2/3}$$

$$T_u = \bigvee_{\nu} P_{\nu u}$$
Hub set selection
Hub set selection
Hub set selection

\[ \omega \text{ s.t. } p(\omega) \min \text{ in } P_{uv}^{2/3} \cap P_{vu}^{2/3} \]

\[ T_u^* \]

\[ T_v^* \]
Hub set selection
Hub set selection

\[ \frac{d(u, w)}{6} = \frac{d(u, y)}{6} \]\n
such that \( w = x_y \)
Hub set selection

\[ \omega \text{ s.t. } P(\omega) \min \text{ in } P_{x\omega w} \alpha \text{ in } P_{wy} \]

\[ \frac{d(u, w)}{6} \]

\[ \frac{d(u, y)}{6} \]

\[ w = x_y \]

\[ T^*_u \]

\[ T^*_v \]
Hub set selection (proof for unweighted graphs)

Draw $\rho(w) \in [0, 1]$ u.a.r. for all $w \in V(G)$.

$H_u = \{w \mid \rho(w) \text{ min. in } P_{x_{ww}}\} \cup \{x_y \mid \rho(x_y) \text{ min. in } P_{x_{yy}}\}$

(Can be computed in $\tilde{O}(n + m)$ separately for each node with shared randomness.)

A sub-path $P_{x_{yy}}$ has length $\frac{d(u, y)}{6}$ and generates a hub in $H_u$ with probability at most $\frac{12}{d(u, y)}$.

$E[|H_u|] \leq \sum_{y \in V(T^*_u)} \frac{12}{d(u, y)} \leq \sum_r |\text{Cut}_r(T^*_u)| \frac{12}{r} = O(k \log D)$
Hub set selection (proof for unweighted graphs)

Draw $\rho(w) \in [0, 1]$ u.a.r. for all $w \in V(G)$.

$H_u = \{w \mid \rho(w) \text{ min. in } P_{xww} \} \cup \{x_y \mid \rho(x_y) \text{ min. in } P_{xyy} \}$

(Can be computed in $\tilde{O}(n + m)$ separately for each node with shared randomness.)

A sub-path $P_{xyy}$ has length $\frac{d(u,y)}{6}$ and generates a hub in $H_u$ with probability at most $\frac{12}{d(u,y)}$.

$E[|H_u|] \leq \sum_{y \in V(T_u^*)} \frac{12}{d(u,y)} \leq \sum_{r} |\text{Cut}_r(T_u^*)| \frac{12}{r} = O(k \log D)$
Hub set selection (proof for unweighted graphs)

Draw $\rho(w) \in [0, 1]$ u.a.r. for all $w \in V(G)$.

$H_u = \{ w \mid \rho(w) \text{ min. in } P_{xww} \} \cup \{ x_y \mid \rho(x_y) \text{ min. in } P_{xyy} \}$

(CanBe computed in $\tilde{O}(n + m)$ separately for each node with shared randomness.)

A sub-path $P_{xyy}$ has length $\frac{d(u,y)}{6}$ and generates a hub in $H_u$ with probability at most $\frac{12}{d(u,y)}$.

$$
E[|H_u|] \leq \sum_{y \in V(T_u^*)} \frac{12}{d(u,y)} \leq \sum_r |\text{Cut}_r(T_u^*)| \frac{12}{r} = O(k \log D)
$$
Road networks: two tree skeletons
What ... maps do?
What ... maps do?
What ...maps do?
Skeleton dimension of grids

\[ k = \Theta(\log n) \]

\[ \mathbf{B}(u, r) \]
Skeleton dimension of grids

$$k = \Theta(\log n)$$
Skeleton dimension of grids

\[ k = \Theta(\log n) \]
Highway dimension $\geq$ skeleton dimension

$$\mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \}$$

$H$ hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_{H \text{ hits } \mathcal{P}_{ur}} |H|$ 

$k \leq h : \text{Cut}_r(T^*_u) \text{ induces a packing in } \mathcal{P}_{ur}, \text{ and } |\text{Cut}_r(T^*_u)| \leq |H|$.
Highway dimension $\geq$ skeleton dimension

$\mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \}$

H hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_H \text{hits } \mathcal{P}_{ur} |H|

k \leq h : \text{Cut}_r(T^*_u) \text{ induces a packing in } \mathcal{P}_{ur}, \text{ and } |\text{Cut}_r(T^*_u)| \leq |H|$.
Highway dimension $\geq$ skeleton dimension

$\mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \}$

$H$ hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_H \text{ hits } \mathcal{P}_{ur} |H|$

$k \leq h : \text{Cut}_r(T_u^*)$ induces a packing in $\mathcal{P}_{ur}$, and $|\text{Cut}_r(T_u^*)| \leq |H|$. 
Highway dimension $\geq$ skeleton dimension

\[ \mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap \overline{B(u, r)} \neq \emptyset \} \]

\( H \) hits \( \mathcal{P}_{ur} \) if \( H \cap P \neq \emptyset \) for all \( P \in \mathcal{P}_{ur} \)

Highway dim. \( h = \max_{u \in \mathcal{P}_{ur}} \min_{H \text{ hits } \mathcal{P}_{ur}} |H| \)

\( k \leq h : \text{Cut}_r(T^*_u) \) induces a packing in \( \mathcal{P}_{ur} \), and \( |\text{Cut}_r(T^*_u)| \leq |H| \).
Highway dimension $\geq$ skeleton dimension

$\mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \}$

$H$ hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_{\text{hits } \mathcal{P}_{ur}} |H|$  

$k \leq h : \text{Cut}_r(T^*_u) \text{ induces a packing in } \mathcal{P}_{ur}, \text{ and } |\text{Cut}_r(T^*_u)| \leq |H|$. 
Highway dimension $\geq$ skeleton dimension

$\mathcal{P}_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \}$

$H$ hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_{H \text{ hits } \mathcal{P}_{ur}} |H|$ \[ k \leq h : \text{Cut}_r(T_u^*) \text{ induces a packing in } \mathcal{P}_{ur}, \text{ and } |\text{Cut}_r(T_u^*)| \leq |H| \]
Highway dimension $\geq$ skeleton dimension

\[ P_{ur} = \{ P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset \} \]

H hits $P_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in P_{ur}$

Highway dim. $h = \max_{ur} \min_{H \text{ hits } P_{ur}} |H|$

$k \leq h : \text{Cut}_r(T_u^*)$ induces a packing in $P_{ur}$, and $|\text{Cut}_r(T_u^*)| \leq |H|$.
Highway dimension $\geq$ skeleton dimension

$$\mathcal{P}_{ur} = \{P \mid |P| > \frac{r}{2} \text{ and } P \cap B(u, r) \neq \emptyset\}$$

$H$ hits $\mathcal{P}_{ur}$ if $H \cap P \neq \emptyset$ for all $P \in \mathcal{P}_{ur}$

Highway dim. $h = \max_{ur} \min_{H \text{ hits } \mathcal{P}_{ur}} |H|$

$k \leq h$ : $\text{Cut}_r(T_u^*)$ induces a packing in $\mathcal{P}_{ur}$, and $|\text{Cut}_r(T_u^*)| \leq |H|$. 
Highway vs skeleton in Brooklyn

Packing of 172 paths

Skeleton width 48
Open: random grid

$k = 70 \quad k = 49 \ (fpp \ [1, 4]) \quad k = 49 \ (prob \ 2/3)$
What about general graphs?
Pre-hub labeling [Angelidakis, Makarychev, Oparin '17]

Hub sets \((H_u)_{u \in V(G)}\) for a graph \(G\) form a pre-hub labeling if for all \(u, v\) pairs, hubs cross on \(P_{uv}\): \(\exists u' \in P_{uv} \cap H_u\) and \(\exists v' \in P_{uv} \cap H_v\) with \(u' \in P_{v'v}\) and \(v' \in P_{uu'}\).

**Theorem**
If shortest paths are unique,
- PHL 2-approximate HL (and pol. time constr.),
- PHL can be converted to HL with \(O(\log D)\) factor.

**Theorem**
If shortest paths are not unique,
- best polyn. time approx. is \(\Omega(\log n)\) (even if \(D = O(1)\)).

**Theorem**
In trees, HHL 2-approximate HL (and pol. time constr.).

**Exercise**: prove it.
Pre-hub labeling [Angelidakis, Makarychev, Oparin ‘17]

Hub sets \((H_u)_{u \in V(G)}\) for a graph \(G\) form a pre-hub labeling if for all \(u, v\) pairs, hubs cross on \(P_{uv} : \exists u' \in P_{uv} \cap H_u\) and \(\exists v' \in P_{uv} \cap H_v\) with \(u' \in P_{v'v}\) and \(v' \in P_{uu'}\).

**Theorem**

If shortest paths are unique,
- PHL 2-approximate HL (and pol. time constr.),
- PHL can be converted to HL with \(O(\log D)\) factor.

**Theorem**

If shortest paths are not unique,
- best polyn. time approx. is \(\Omega(\log n)\) (even if \(D = O(1)\)).

**Theorem**

In trees, HHL 2-approximate HL (and pol. time constr.).

**Exercise**: prove it.
Pre-hub labeling [Angelidakis, Makarychev, Oparin '17]

Hub sets \((H_u)_{u \in V(G)}\) for a graph \(G\) form a pre-hub labeling if for all \(u, v\) pairs, hubs cross on \(P_{uv}: \exists u' \in P_{uv} \cap H_u\) and \(\exists v' \in P_{uv} \cap H_v\) with \(u' \in P_{v'v}\) and \(v' \in P_{uu'}\).

**Theorem**

If shortest paths are unique,
- PHL 2-approximate HL (and pol. time constr.),
- PHL can be converted to HL with \(O(\log D)\) factor.

**Theorem**

If shortest paths are not unique,
- best polyn. time approx. is \(\Omega(\log n)\) (even if \(D = O(1)\)).

**Theorem**

In trees, HHL 2-approximate HL (and pol. time constr.).

**Exercise**: prove it.
Pre-hub labeling [Angelidakis, Makarychev, Oparin '17]

Hub sets \((H_u)_{u \in V(G)}\) for a graph \(G\) form a pre-hub labeling if for all \(u, v\) pairs, hubs cross on \(P_{uv}\): \(\exists u' \in P_{uv} \cap H_u\) and \(\exists v' \in P_{uv} \cap H_v\) with \(u' \in P_{v'v}\) and \(v' \in P_{uu'}\).

**Theorem**
If shortest paths are unique,
- PHL 2-approximate HL (and pol. time constr.),
- PHL can be converted to HL with \(O(\log D)\) factor.

**Theorem**
If shortest paths are not unique,
- best polyn. time approx. is \(\Omega(\log n)\) (even if \(D = O(1)\)).

**Theorem**
In trees, HHL 2-approximate HL (and pol. time constr.).

**Exercise** : prove it.
Pre-hub labeling [Angelidakis, Makarychev, Oparin '17]

Hub sets \((H_u)_{u \in V(G)}\) for a graph \(G\) form a pre-hub labeling if for all \(u, v\) pairs, hubs cross on \(P_{uv}\): \(\exists u' \in P_{uv} \cap H_u\) and \(\exists v' \in P_{uv} \cap H_v\) with \(u' \in P_{v'v}\) and \(v' \in P_{uu'}\).

**Theorem**
If shortest paths are unique,
- PHL 2-approximate HL (and pol. time constr.),
- PHL can be converted to HL with \(O(\log D)\) factor.

**Theorem**
If shortest paths are not unique,
- best polyn. time approx. is \(\Omega(\log n)\) (even if \(D = O(1)\)).

**Theorem**
In trees, HHL 2-approximate HL (and pol. time constr.).

**Exercise**: prove it.
Hub labeling of general graphs

Any graph has $O\left(\frac{n}{\log n}\right)$ hubsets for $m = O(n)$ (combining [Kosowski, V. ‘17] and [Alstrup et al ‘16]).

Idea (for $\Delta = O(1)$):

- for $r \geq \delta$, the width of a $r$-cut of a skeleton tree is $k = O(n/\delta)$ (we can get $O(n/\delta)$ hubsets for distances $\geq \delta$),
- select as hubs all nodes at distance less than $\delta = \frac{\log n}{2 \log \Delta}$ (at most $\sqrt{n}$ nodes).
Hub labeling of general graphs

Any graph has $O\left(\frac{n}{\log n}\right)$ hubsets for $m = O(n)$ (combining [Kosowski, V. ‘17] and [Alstrup et al ‘16]).

Idea (for $\Delta = O(1)$):
- for $r \geq \delta$, the width of a $r$-cut of a skeleton tree is $k = O(n/\delta)$ (we can get $O(n/\delta)$ hubsets for distances $\geq \delta$),
- select as hubs all nodes at distance less than $\delta = \frac{\log n}{2 \log \Delta}$ (at most $\sqrt{n}$ nodes).
Theorem [Kosowski, Uznański, V. ‘19]: There exists graphs with max-degree 3 such that any hubsets have average size \( \Omega\left( \frac{n}{2^{O(\sqrt{\log n})}} \right) \).

Linked to Ruzsa-Szemerédi function bounding the number of edges in a graph decomposable into \( n \) induced matchings.

There exists graphs with max-degree 3 such that any distance labels must have average size \( \frac{\text{SUMINDEX}(n)}{2^{O(\sqrt{\log n})}} \).

Open problem: does any sparse graph have a (centralized) distance oracle of size \( O(n^{1.5}) \) and query time \( O(n^{0.5}) \)?
Lower bound on hub labeling of general graphs

Theorem [Kosowski, Uznański, V. ’19]: There exists graphs with max-degree 3 such that any hubsets have average size \( \Omega\left(\frac{n}{2^{O(\sqrt{\log n})}}\right) \).

Linked to Ruzsa-Szemerédi function bounding the number of edges in a graph decomposable into \( n \) induced matchings.

There exists graphs with max-degree 3 such that any distance labels must have average size \( \frac{\text{SUMINDEX}(n)}{2^{O(\sqrt{\log n})}} \).

Open problem: does any sparse graph have a (centralized) distance oracle of size \( O(n^{1.5}) \) and query time \( O(n^{0.5}) \)?
Lower bound on hub labeling of general graphs

**Theorem** [Kosowski, Uznański, V. ’19] : There exists graphs with max-degree 3 such that any hubsets have average size $\Omega\left(\frac{n}{2^{O(\sqrt{\log n})}}\right)$.

Linked to Ruzsa-Szemerédi function bounding the number of edges in a graph decomposable into $n$ induced matchings.

There exists graphs with max-degree 3 such that any distance labels must have average size $\frac{\text{SUMINDEX}(n)}{2^{O(\sqrt{\log n})}}$.

Open problem : does any sparse graph have a (centralized) distance oracle of size $O(n^{1.5})$ and query time $O(n^{0.5})$?
Lower bound on hub labeling of general graphs

**Theorem** [Kosowski, Uznański, V. ‘19]: There exists graphs with max-degree 3 such that any hubsets have average size $\Omega\left(\frac{n}{2^{O(\sqrt{\log n})}}\right)$.

Linked to Ruzsa-Szemerédi function bounding the number of edges in a graph decomposable into $n$ induced matchings.

There exists graphs with max-degree 3 such that any distance labels must have average size $\frac{\text{SUMINDEX}(n)}{2^{O(\sqrt{\log n})}}$.

**Open problem**: does any sparse graph have a (centralized) distance oracle of size $O(n^{1.5})$ and query time $O(n^{0.5})$?
HL hard instance: $2\ell + 1$ grids of dim. $\ell = \sqrt{\log n}$

$G_X = G \setminus \{ y_\ell \mid X_y = 0 \}$, send $x = 2a, L_{x_0}, z = 2b, L_{z_{2\ell}}$, check $d(x_0, z_{2\ell})$. 

$V_{2\ell}$

$V_\ell$

$V_0$

(1,0)

(2,1)

(3,2)
What about more hops?
Given a connected weighted undirected graph $G$, we define the \textbf{h-hop distance} $d^h_G(u, v)$ between $u$ and $v$ as the minimum weight of a path from $u$ to $v$ with $h$ edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most $h$ hops.) The usual distance from $u$ to $v$ is thus $d_G(u, v) = d^{n-1}_G(u, v)$. We define a \textbf{h-hopset of $G$} as a set $H$ of edges such that $d^h_{G \cup H}(u, v) = d_G(u, v)$ where each edge $uv$ of $H$ is considered to have weight $d^h_G(u, v)$.

(a) What is the minimum number of edges in $G \cup H$ when $H$ is a 1-hopset of $G$?

(b) What notion seen in course is tightly related to the notion of 2-hopset?

(c) Suppose that $G$ is a path of length $n$, propose a 3-hopset of $G$ with as few edges as you can (we do not care about multiplicative constants).

(d) Same question for a 4-hopset.

(e) Consider a 3-hopset $H$ of a graph $G$. Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when $G$ is a path of length $n$?
Given a connected weighted undirected graph \( G \), we define the \( h \)-hop distance \( d^h_G(u, v) \) between \( u \) and \( v \) as the minimum weight of a path from \( u \) to \( v \) with \( h \) edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most \( h \) hops.) The usual distance from \( u \) to \( v \) is thus \( d_G(u, v) = d^{n-1}_G(u, v) \). We define a \( h \)-hopset of \( G \) as a set \( H \) of edges such that \( d^h_{G \cup H}(u, v) = d_G(u, v) \) where each edge \( uv \) of \( H \) is considered to have weight \( d_G(u, v) \).

(a) What is the minimum number of edges in \( G \cup H \) when \( H \) is a 1-hopset of \( G \)?

(b) What notion seen in course is tightly related to the notion of 2-hopset?

(c) Suppose that \( G \) is a path of length \( n \), propose a 3-hopset of \( G \) with as few edges as you can (we do not care about multiplicative constants).

(d) Same question for a 4-hopset.

(e) Consider a 3-hopset \( H \) of a graph \( G \). Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when \( G \) is a path of length \( n \)?
Given a connected weighted undirected graph \( G \), we define the \( h \)-hop distance \( d^h_G(u, v) \) between \( u \) and \( v \) as the minimum weight of a path from \( u \) to \( v \) with \( h \) edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most \( h \) hops.) The usual distance from \( u \) to \( v \) is thus \( d_G(u, v) = d^{n-1}_G(u, v) \). We define a \( h \)-hopset of \( G \) as a set \( H \) of edges such that \( d^h_{G \cup H}(u, v) = d_G(u, v) \) where each edge \( uv \) of \( H \) is considered to have weight \( d_G(u, v) \).

(a) What is the minimum number of edges in \( G \cup H \) when \( H \) is a 1-hopset of \( G \)?

(b) What notion seen in course is tightly related to the notion of 2-hopset?

(c) Suppose that \( G \) is a path of length \( n \), propose a 3-hopset of \( G \) with as few edges as you can (we do not care about multiplicative constants).

(d) Same question for a 4-hopset.

(e) Consider a 3-hopset \( H \) of a graph \( G \). Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when \( G \) is a path of length \( n \)?
Given a connected weighted undirected graph $G$, we define the h-hop distance $d^h_G(u,v)$ between $u$ and $v$ as the minimum weight of a path from $u$ to $v$ with $h$ edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most $h$ hops.) The usual distance from $u$ to $v$ is thus $d_G(u,v) = d^{n-1}_G(u,v)$. We define a h-hopset of $G$ as a set $H$ of edges such that $d^h_{G\cup H}(u,v) = d_G(u,v)$ where each edge $uv$ of $H$ is considered to have weight $d_G(u,v)$.

(a) What is the minimum number of edges in $G \cup H$ when $H$ is a 1-hopset of $G$?

(b) What notion seen in course is tightly related to the notion of 2-hopset?

(c) Suppose that $G$ is a path of length $n$, propose a 3-hopset of $G$ with as few edges as you can (we do not care about multiplicative constants).

(d) Same question for a 4-hopset.

(e) Consider a 3-hopset $H$ of a graph $G$. Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when $G$ is a path of length $n$?
Given a connected weighted undirected graph $G$, we define the $h$-hop distance $d^h_G(u, v)$ between $u$ and $v$ as the minimum weight of a path from $u$ to $v$ with $h$ edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most $h$ hops.) The usual distance from $u$ to $v$ is thus $d_G(u, v) = d^{n-1}_G(u, v)$. We define a $h$-hopset of $G$ as a set $H$ of edges such that $d^h_{G \cup H}(u, v) = d_G(u, v)$ where each edge $uv$ of $H$ is considered to have weight $d_G(u, v)$.

(a) What is the minimum number of edges in $G \cup H$ when $H$ is a 1-hopset of $G$?
(b) What notion seen in course is tightly related to the notion of 2-hopset?
(c) Suppose that $G$ is a path of length $n$, propose a 3-hopset of $G$ with as few edges as you can (we do not care about multiplicative constants).
(d) Same question for a 4-hopset.
(e) Consider a 3-hopset $H$ of a graph $G$. Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when $G$ is a path of length $n$?
Given a connected weighted undirected graph $G$, we define the $h$-hop distance $d^h_G(u, v)$ between $u$ and $v$ as the minimum weight of a path from $u$ to $v$ with $h$ edges at most. (Each edge of a path is called a “hop”, it thus corresponds to the distance using at most $h$ hops.) The usual distance from $u$ to $v$ is thus $d_G(u, v) = d^{n-1}_G(u, v)$. We define a $h$-hopset of $G$ as a set $H$ of edges such that $d^h_{G∪H}(u, v) = d_G(u, v)$ where each edge $uv$ of $H$ is considered to have weight $d_G(u, v)$.

(a) What is the minimum number of edges in $G∪H$ when $H$ is a 1-hopset of $G$?

(b) What notion seen in course is tightly related to the notion of 2-hopset?

(c) Suppose that $G$ is a path of length $n$, propose a 3-hopset of $G$ with as few edges as you can (we do not care about multiplicative constants).

(d) Same question for a 4-hopset.

(e) Consider a 3-hopset $H$ of a graph $G$. Propose a distance oracle based on distinguishing middle edges of 3-hop shortest paths from the two others. What query time do you obtain when $G$ is a path of length $n$?
3-Hopsets in Graphs with Bounded Skeleton Dimension

Theorem (Gupta, Kosowski, V. 2019)

For a unique shortest path graph with skeleton dimension $k$ and polylog average link length, there exists a randomized construction of a $3$-hopset distance oracle of size $|H| = O(n k \log k \log \log n)$, which for an arbitrary queried node pair performs distance queries in expected time $O(k^2 \log^2 k \log^2 \log n)$ (where the expectation is taken over the randomized construction of the oracle).