Graph Inference with Distance Queries

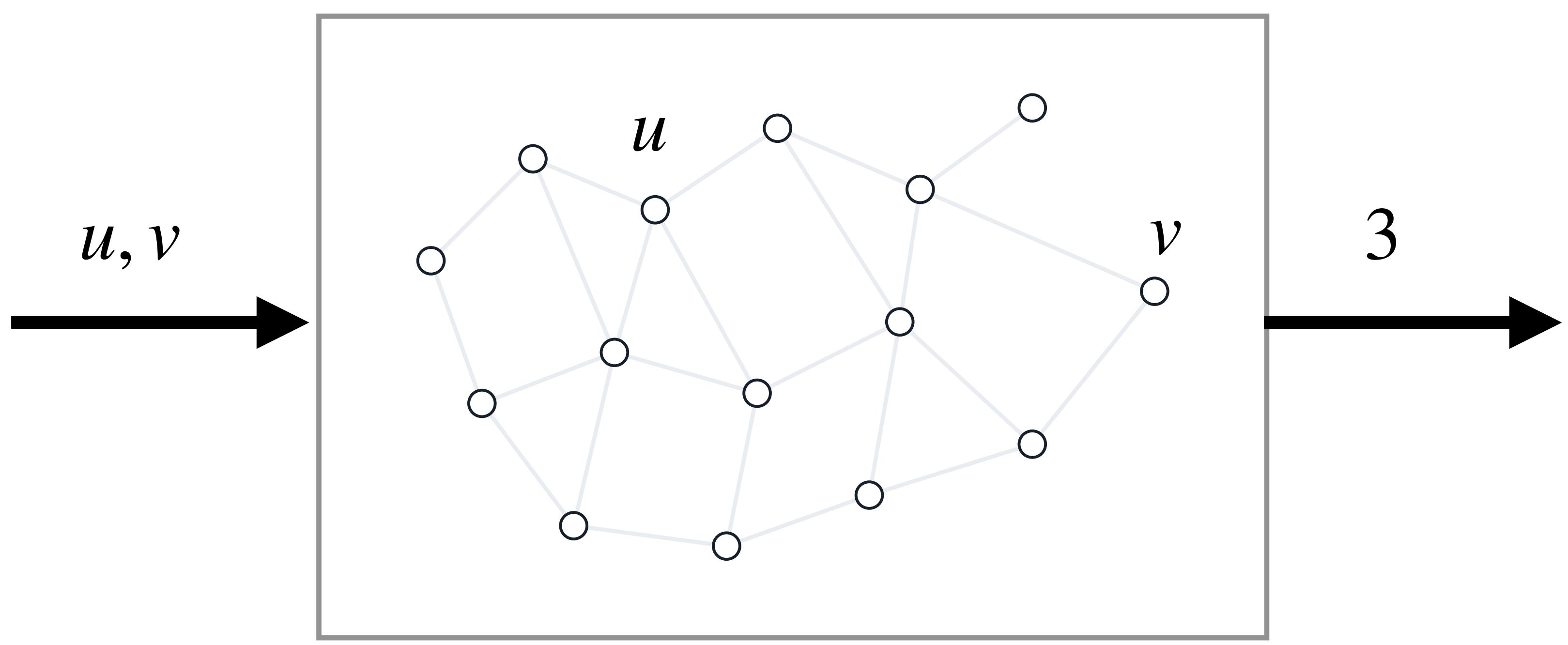
Amir Nayyeri

Oregon State University

Joint work with Mitchell Black, Huck Bennett, Chirag Kaudan, Evelyn Warton

Graph Reconstruction with Distance Queries

- Given:
  - Graph $G = (V, E)$, with known vertices and hidden edges
  - Graph Distance Oracle: Given $u, v \in V$, returns $d(u, v)$



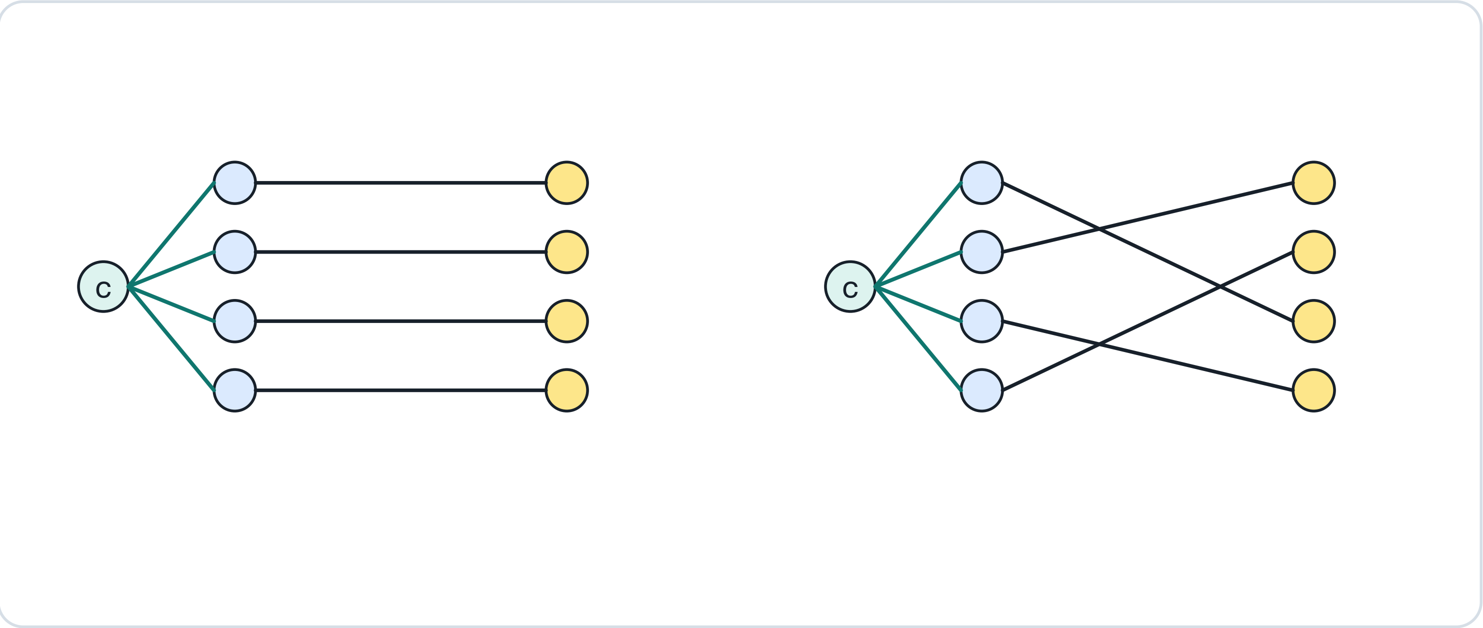
- **Problem:** How many distance queries we need to reconstruct G?

$O(n^2)$ Upper Bound for Reconstruction

- Check for and edge between every pair of vertices
  - There is an edge iff the shortest path length is one

$\Omega(n^2)$ Lower Bounds

- Extremely sparse (disconnected) graphs; e.g. a single edge [Reyzin Srivastava 07]
- Trees with high-degree [Reyzin Srivastava 07]
  - Hard to reconstruct a subdivided star even with fixed center, first layer and second layer. (reduction from learning matchings with edge detection queries [Alon Beigel Kasif Rudich Sudakov 04])



Subquadratic Reconstruction with SP queries

- Bounded degree trees and tree-like graphs: $O(n \log n)$ queries

- Bounded bfs-width graphs
  - Bounded bandwidth: $n \log^{O(1)} n$ queries
  - Bounded degree bounded pathwidth: $O(n^{2-\varepsilon})$

- General bounded degree graphs: $\widetilde{O}(n^{3/2})$-queries with high prob.

Subquadratic Reconstruction with SP queries

- Bounded degree trees and tree-like graphs: $O(n \log n)$ queries

- Bounded bfs-width graphs
  - Bounded bandwidth: $n \log^{O(1)} n$ queries
  - Bounded degree bounded pathwidth: $O(n^{2-\varepsilon})$

- General bounded degree graphs: $\widetilde{O}(n^{3/2})$-queries with high prob.

Subquadratic Reconstruction with SP queries

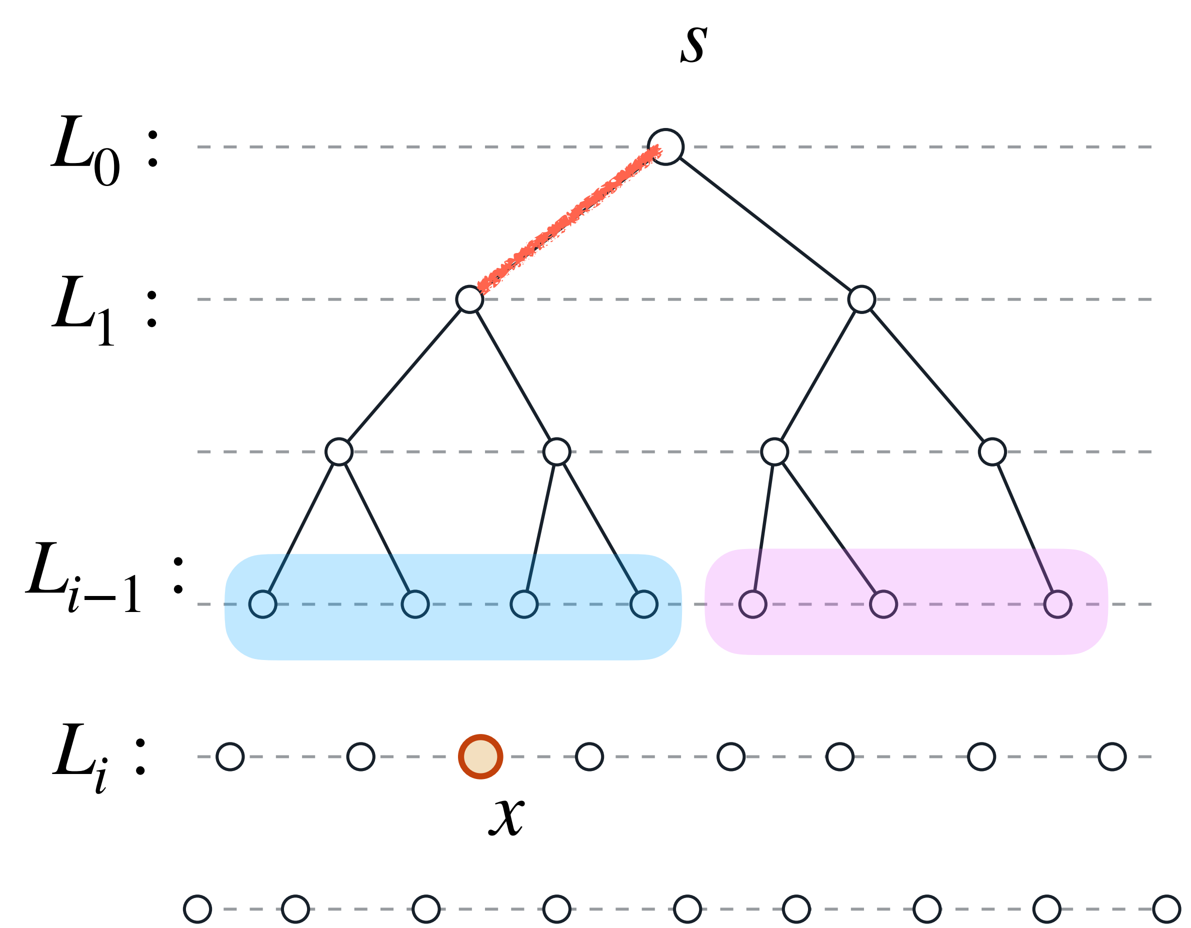
- **Bounded degree trees and tree-like graphs:** $O(n \log n)$ **queries**

- Bounded bfs-width graphs
  - Bounded bandwidth: $n \log^{O(1)} n$ queries
  - Bounded degree bounded pathwidth: $O(n^{2-\varepsilon})$

- General bounded degree graphs: $\widetilde{O}(n^{3/2})$-queries with high prob.

Bounded degree trees — $\Theta(n \log n)$

- Pick a root $s \in V$
- Build bfs layers from s ($n - 1$ queries)
- Traverse vertices layer by layer
- Find the parent of each vertex:
  - Use tree balanced separator to binary search among L_{i-1} ($O(\log n)$ queries per vertex)



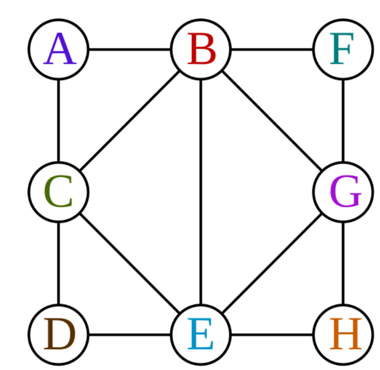
[Hein 89, Bestide Groenland 24]

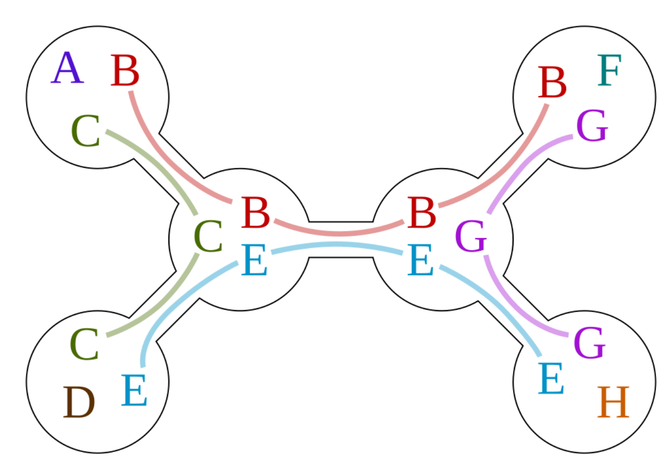
Tree decomposition, length and width

Tree decomposition of $G = (V, E)$: bags of vertices in V arranged on a tree T such that

- Each vertex is contained in a non-empty connected subtree of T
- Each edge is contained in at least one bag

- Length of a decomposition: the max diam of its bags
  - **Treelength**: min length over all tree decompositions
- Width of a decomposition: the max size of its bags (minus one)
  - **Treewidth**: min width over all tree decompositions





Tree decomposition, length and width

Treelength and treewidth are very different parameters:

- K_n has treelength 1 and treewidth $n-1$
- C_n has treelength $n/3$ and treewidth 2

Useful intuition: A graph has bounded treelength if and only if its shortest-path metric embeds into a tree metric with bounded additive distortion [Berger Seymour 24]

Tree-like graphs

Bounded degree bounded treelength graphs can be reconstructed (deterministically) with $O(n \log n)$ shortest paths queries [Kaudan N '25]

- **Intuition:** If I know the graph up to layer i I can approximate optimal decomposition use to layer $i - \delta$ with $\delta = O(1)$.

To find parents of vertices in L_i

- Binary search on partially build decomposition up to layer $i - \delta$
- Exhaustive search on the remaining candidates

Tree-like graphs

Bounded degree bounded treelength graphs can be reconstructed (deterministically) with $O(n \log n)$ shortest paths queries [Kaudan N '25]

- chordal $\subseteq$ k-chordal $\subseteq$ bounded treelength
  - Sequence of work on graph reconstruction with SP queries for these families [Kannan Mathieu Zhou 2018][Rong Li Yang Wang 2021][Bestide Groenland 24a] [Bestide Groenland 24a]

Open: What about bounded degree bounded treewidth graphs?

Subquadratic Reconstruction with SP queries

- Bounded degree trees and tree-like graphs: $O(n \log n)$ queries

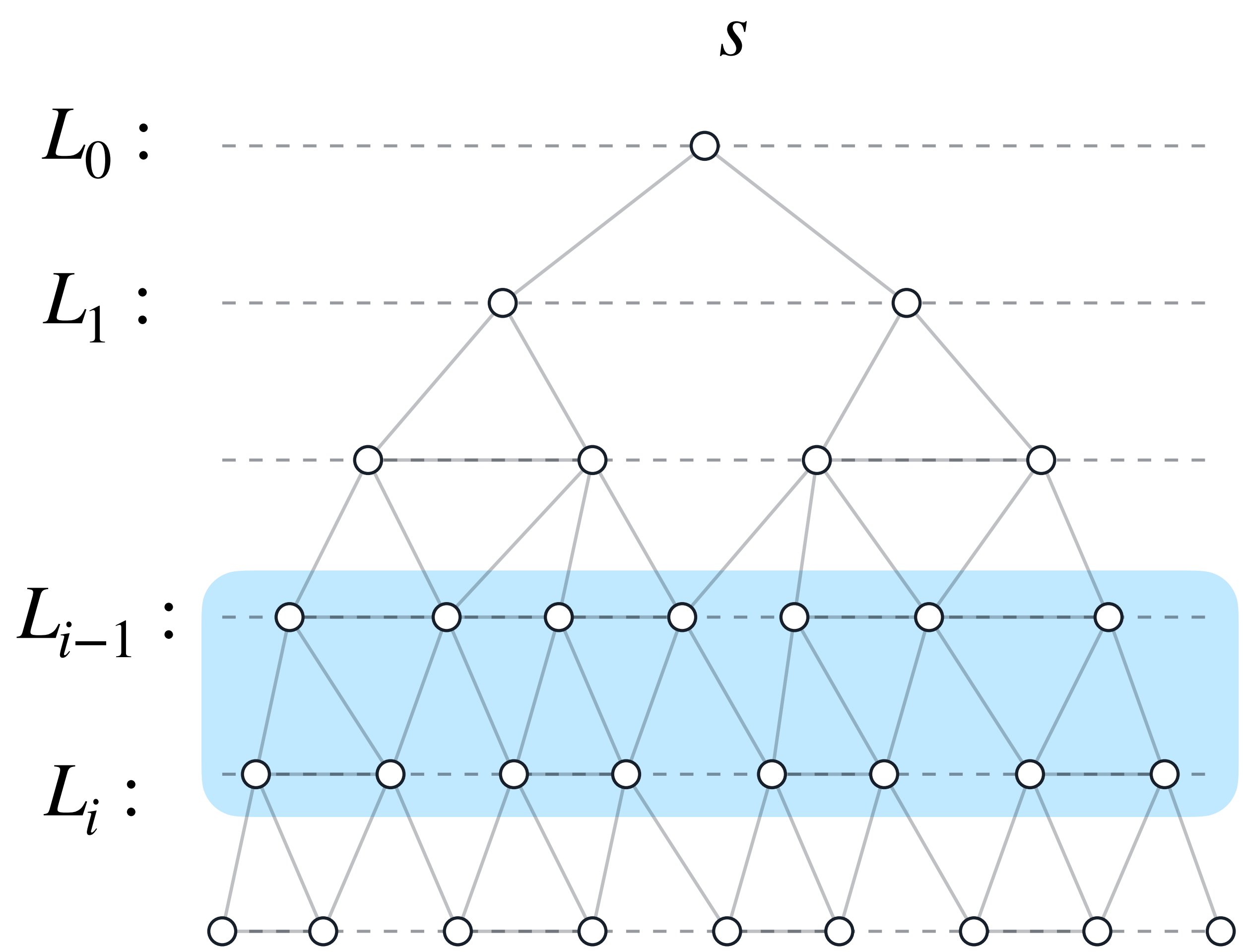
- **Bounded bfs-width graphs**
  - Bounded bandwidth: $n \log^{O(1)} n$ queries
  - Bounded degree bounded pathwidth: $O(n^{2-\varepsilon})$

- General bounded degree graphs: $\widetilde{O}(n^{3/2})$-queries with high prob.

Graphs with bounded bfs-width

Simpler Algorithm:

- Pick a root $s \in V$
- Build bfs layers ($n - 1$ queries)
- Check all ``potential" edges; edges with endpoints in some $S_i = L_{i-1} \cup L_i$



- **Query complexity:** $O(bn)$, b widest bfs layer

Graphs with bounded bfs-width

- Bounded bandwidth graphs have bfs-width $\log^{O(1)} n$ [Eppstein Goodrich Liu 25] $\Rightarrow O(n \log^{O(1)} n)$ reconstruction algorithm

- Bounded degree bounded pathwidth graphs have bfs-width $n^{1-\varepsilon}$ [Kaudan N 26] $\Rightarrow O(n^{2-\varepsilon})$ reconstruction algorithm.

Open: What about bounded degree bounded treewidth graphs?

Subquadratic Reconstruction with SP queries

- Bounded degree trees and tree-like graphs: $O(n \log n)$ queries

- Bounded bfs-width graphs
  - Bounded bandwidth: $n \log^{O(1)} n$ queries
  - Bounded degree bounded pathwidth: $O(n^{2-\varepsilon})$

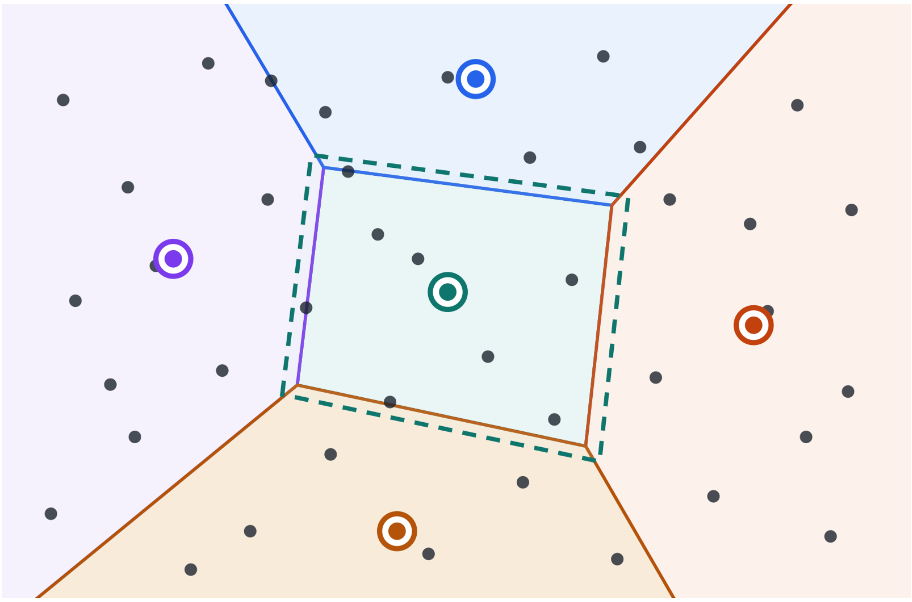
- **General bounded degree graphs: $\widetilde{O}(n^{3/2})$-queries with high prob.**

Bounded degree graphs — $\widetilde{O}(n^{3/2})$

- Abstract algorithm:
  - Build a covering of edges, $S_1, \ldots, S_k \subseteq V$,
    - Each edge has its both endpoints in at least one of S_i's
    - For example: S_i's can be consecutive pairs of bfs layers
  - Reconstruct the induced graph of each S_i with brute force, in quadratic time
- Reconstruction time: $\sum |S_i|^2$
- A covering with $k = \widetilde{O}(\sqrt{n})$ and $|S_i| = O(\sqrt{n})$ can be computed with $\widetilde{O}(n^{3/2})$ queries for bounded degree graphs $\Rightarrow$ Total $\widetilde{O}(n^{3/2})$ query complexity [Kannan Mathieu Zhou 18]

Building the covering [Kannan Mathieu Zhou '18]

- $S_1, \ldots, S_k$ are extended Voronoi cells of a set A of ``landmarks".

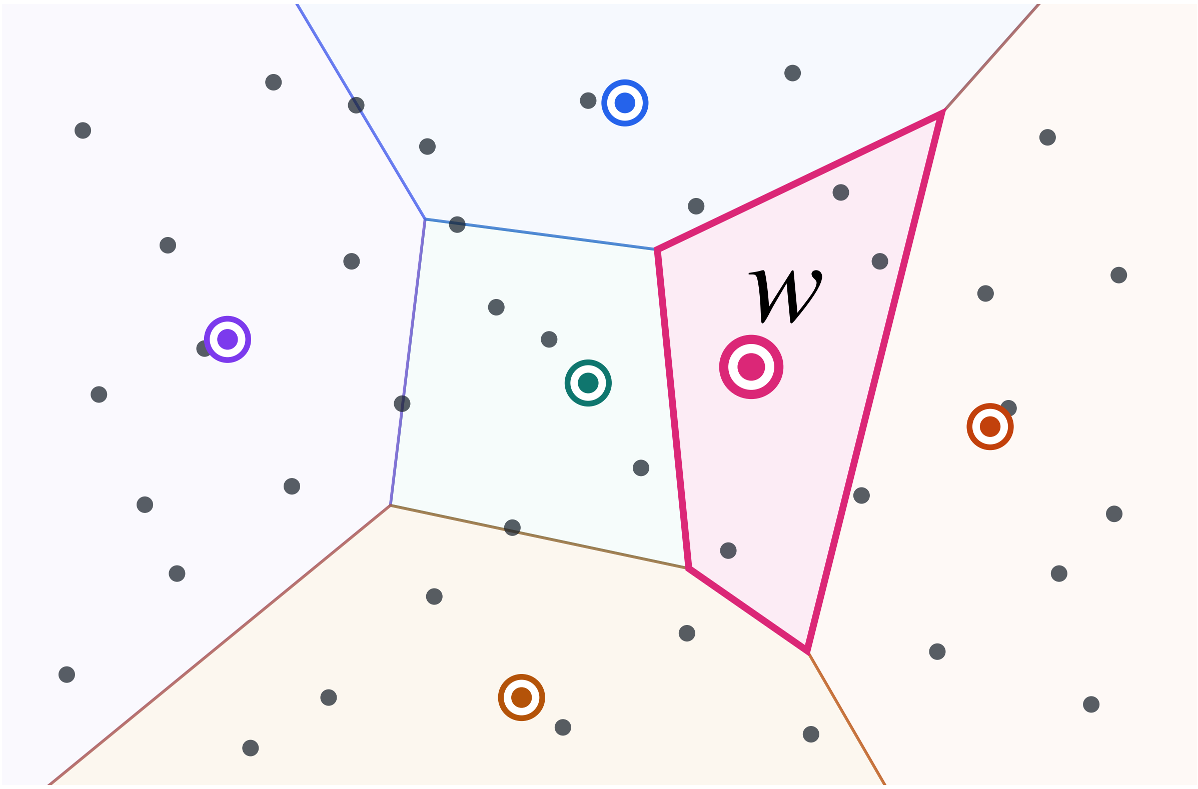


- Extended Voronoi cells: include into each Voronoi cell its one hop neighbors
- $\Rightarrow$ all edges are covered

Building the covering [Kannan Mathieu Zhou 18]

- A is computed via rounds of sampling
  - $W \subseteq V$: is the set of vertices w with ``large clusters"
    - $|\{u \in V \,|\, d(u,w) < d(u,A)\}| > \sqrt{n}$
    - The Voronoi cell of w in the Voronoi diagram of $\{w\} \cup A$ is large ($> \sqrt{n}$)
  - Add each $w \in W$ to A with probability ~$1/\sqrt{n}$

Similar to [Thorup Zwick 01]



Reconstruction with shortest path queries:

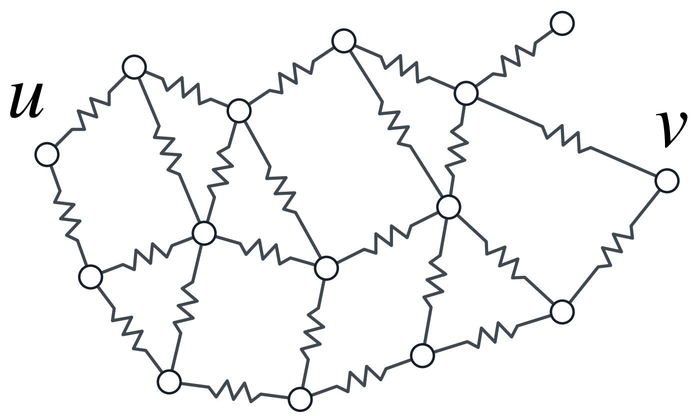
	Number of Queries
General	$\Theta(n^2)$
Bounded deg bounded treelength	$\Theta(n \log n)$
Bounded bandwidth	$n \log^{O(1)} n$
Bounded deg bounded pathwidth (~bounded cutwidth)	$O(n^{2-\varepsilon})$
Bounded deg (randomized)	$O(n^{3/2})$

Shortest path vs. effective resistance queries:

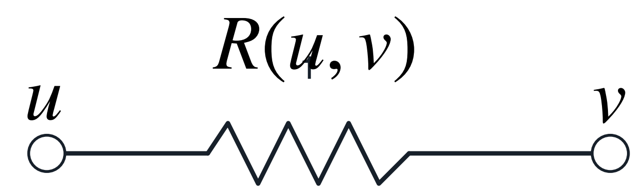
	Number of SP Queries	Number of ER queries
General	$\Theta(n^2)$	$\Theta(n^2)$
Bounded deg bounded treelength	$\Theta(n \log n)$	?
Bounded bandwidth	$n \log^{O(1)} n$	?
Bounded deg bounded pathwidth (~bounded cutwidth)	$O(n^{2-\varepsilon})$	?
Bounded deg (randomized)	$O(n^{3/2})$	?

Effective resistance

- Effective resistance distance, $R(u, v)$
  - The resistance of the equivalent one-resistor network for u, v



$\Leftrightarrow$

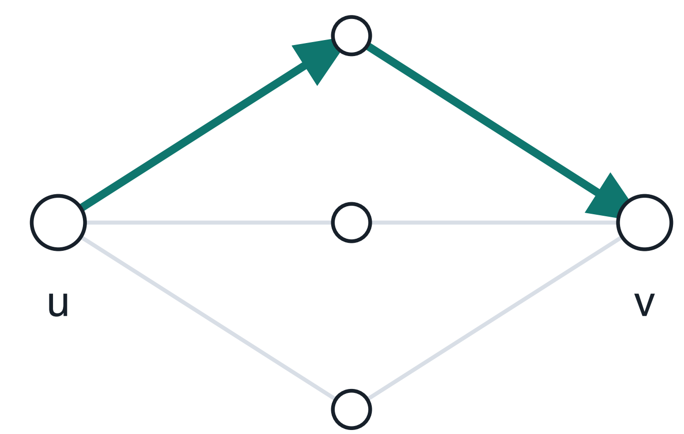


- $R(u, v) = \dfrac{C_{u,v}}{2m}$, $C_{u,v}$: commute time between u, v (expected time for a random walk starting at u to visit v and come)

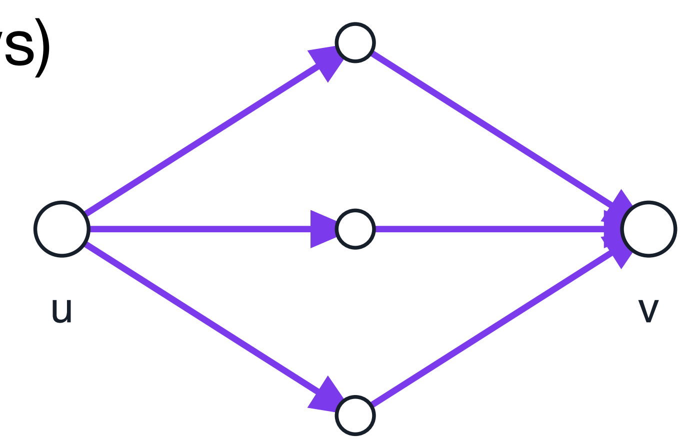
- $R(u, v) = (e_u - e_v)^T L_G^+ (e_u - e_v)$, $L_G = D_G - A_G$ is Laplacian of G

SP and ER, flow interpretation

- Shortest path distance
  - $d(u, v)$: Minimum ℓ_1-value of a u to v unit flow
  - $d(u, v) = \min \|f\|_1$ (f ranges over u to v unit flows)



- Effective resistance distance,
  - $R(u, v)$: Minimum squared ℓ_2-value of a u to v flow
  - $R(u, v) = \min \|f\|_2^2$ (f ranges over u to v unit flows)



$\Theta(n^2)$ ER queries for general graphs

- $O(n^2)$-query algorithm [Spielman 12, Hoskin Musco Musco Tsourakakis 18]
  - Interesting formula: $L^+ = -\frac{1}{2}\Pi R \Pi$
    - $\Pi = I - \frac{1}{n} \cdot 11^T$ (Project away from all 1 vector, the kernel of Laplacian)
    - R: The matrix of all pairwise effective resistances

- $\Omega(n^2)$ lower bound is implied by shortest path lower bounds
  - ER and SP are the same on acyclic graphs

Big Open question:

Which graph families beyond bounded degree trees admit reconstruction with a subquadratic number of ER queries?

- Testing some properties is possible with linear ER queries
- Graph completion is possible with linear queries: given a graph with k missing adjacency matrix queries, we can complete it with k ER queries

[Black Bennett N Warton ‘26]

What is missing compared to shortest paths?

Methods for finding the covering set

- No bfs-layer property
- No obvious ER analogue of the Voronoi covering construction

Methods to reconstruct an induced subgraph using number of queries proportional to its size:

- In particular, no obvious way to check if there is an edge between two vertices quickly
- In fact, a linear lower bound for this task!

Discovering the neighbors of a vertex

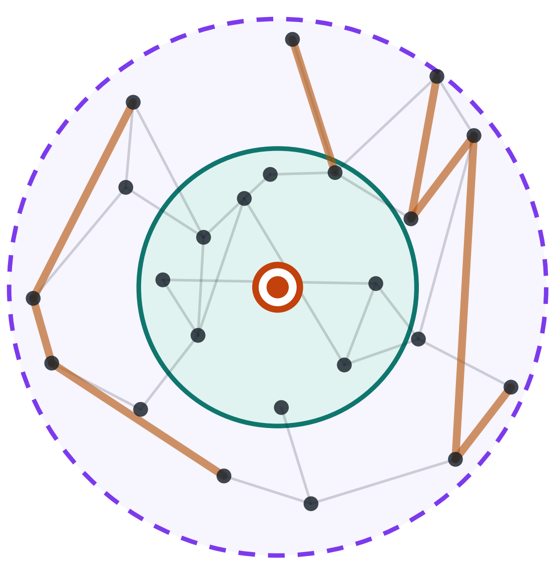
The neighbors of a vertex v can be found with $n - 1 + |B_R(v,2)|^2$ ER queries.

- $B_R(v,2)$ is the set of all vertices that are at ER distance at most 2 from v.

Discovering the neighbors of a vertex

- Algorithm:
  - ER-query all vertices from v to compute $B_R(v,2)$
  - ER-query all pairs within $B_R(v,2)$
  - Reconstruct the **weighted** graph H on $B_R(v,2)$ that realize ER values between these vertices in G (using $|B_R(v,2)|^2$ ER queries)
    - H: Schur complement of G onto $B_R(v,2)$
  - Return neighbors of v in H

G



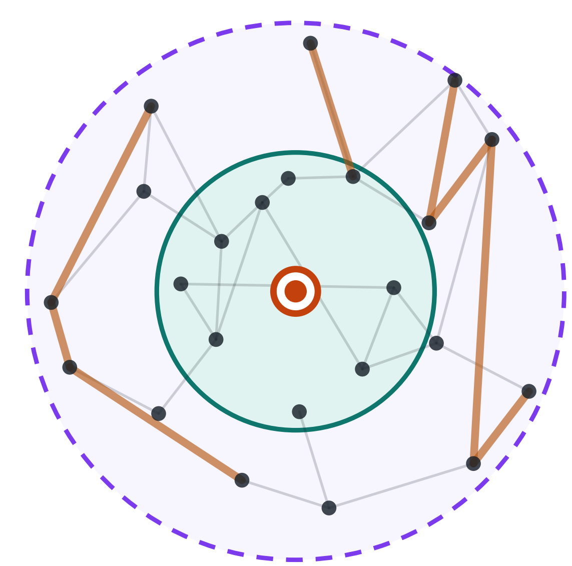
H

Discovering the neighbors of a vertex

Neighbors of v in H and G are identical

- $uv \in E \Rightarrow R(u, v) \leq 1$
- Algorithm to build Schur complement onto $S \subseteq V$:
    - Iteratively, eliminate every vertex $v \in V \backslash S$ by replacing it with a clique on its neighbors (with appropriate weights)
- The process does not change the weight of an edge that entirely in the smaller ball $B(v,1)$, as outside vertices don't have neighbors in the smaller ball
- All neighbors x of v are in $B(v,1)$; $R(v, x) \leq 1$

G



H

How dense is ER space?

The neighbors of a vertex v can be found with $n - 1 + |B_R(v,2)|^2$ ER queries.

- How large is $B_R(v,2)$?
  - As large as the entire graph (expanders)
  - Constant size for bounded deg bounded treewidth and bounded deg planar graphs

- **Implication:** Property testers with $f(\varepsilon, n)$ queries in bounded degree model can be simulated with $n \cdot f(\varepsilon, n)$ ER queries in bounded treewidth and planar graphs. Examples: k-connectivity, bipartiteness, presence of long cycles

Thanks/Open Problems!

	Number of SP Queries	Number of ER queries
General	$\Theta(n^2)$	$\Theta(n^2)$
Bounded deg bounded treelength	$\Theta(n \log n)$	?
Bounded bandwidth	$n \log^{O(1)} n$	?
Bounded deg bounded pathwidth (~bounded cutwidth)	$O(n^{2-\varepsilon})$	?
Bounded deg (randomized)	$O(n^{3/2})$	?