Distributed Distance-Bounded Network Design Through Distributed Convex Programming

Workshop on Local Algorithms (appeared in OPODIS 2017).

Michael Dinitz, Yasamin Nazari
Johns Hopkins University

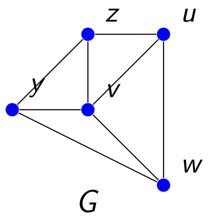
June, 2018

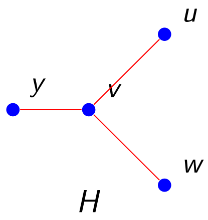
Distance Bounded Network Design

- Given input graph G, find a subgraph H with minimum cost such that certain pairs of vertices are within some distance bound of each other in H of at most D.
- A well-known class are *graph spanners*: the distance in H for certain pairs is within a certain factor of their original distance in G.
- Most of these problems are NP-hard so the focus is often polynomial time approximation algorithms.

Example of Distance Bounded Network Design

- Set of demands: $S = \{(u,v),(w,v),(y,w)\}$, distance bound $D = 2$.
- Find smallest subgraph H of G s.t, all the pairs in S are connected with a path of length at most 2.





Distributed Linear Programming

- These problems are often modeled by a Linear Program.
- Solving linear programs is often challenging task in distributed settings.

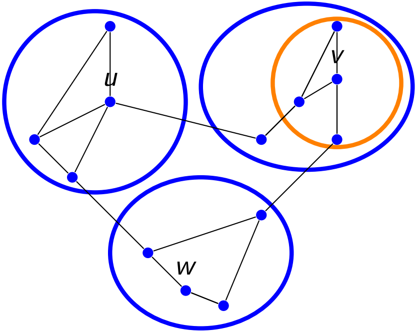
We provide efficient distributed algorithms for distance-bounded network design LPs and a class of convex programs that generalize these problems in the $\mathcal{LOCAL}$ model.

Summary of Results

- For several distance-bounded network design problems, we give approximation guarantees that match their best known centralized bounds.
- Example of these problems are: Directed k-Spanner, Basic 3-Spanner, Basic 4-Spanner and Lowest-Degree k-Spanner.
- These algorithms run in $O(D \log n)$ rounds, where D is the maximum distance bound.
- This is the best known bound for these problems in the distributed $\mathcal{LOCAL}$ model when the local computation is polynomial time.

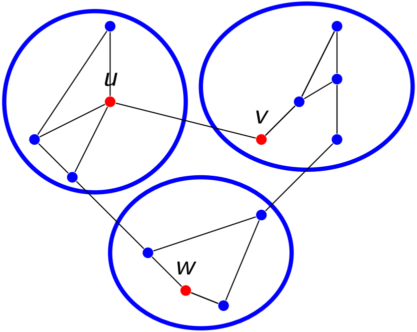
(k, ϵ)-Padded Decomposition

- Every cluster has low diameter of $O((k/\epsilon) \log n)$.
- For each node the probability that all nodes in its k-neighborhood are in the same cluster is at least $1 - \epsilon$.



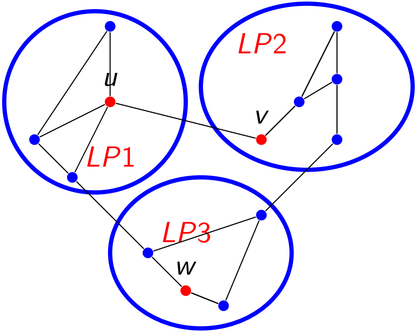
High Level Idea: Partition

- Partition the graph by a distributed algorithm that samples from a (D, ϵ)-padded decomposition in $O(\frac{D}{\epsilon} \ln n)$ rounds.
- Nodes know the center of the cluster they belong to.



High Level Idea: Solving Local LPs

- The center of each cluster solves a local linear program.
- Cluster center broadcasts the solutions to all the nodes in the cluster.



High Level Idea: Putting it together

- Repeat this process $O(\frac{\ln n}{\epsilon})$ times in *parallel* (decompositions are independent).
- For each edge taking average over local solutions for iterations in which the ball around that edge is in the same cluster will yield to a global solution.
- Using Chernoff bounds, we show that the global solution formed is feasible to the global LP and is a constant factor of the optimal solution.

- Thanks!
- Questions? Comments? Please come see my poster!