Distributed Distance-Bounded Network Design Through Distributed Convex Programming

OPODIS 2017

Michael Dinitz, Yasamin Nazari
Johns Hopkins University

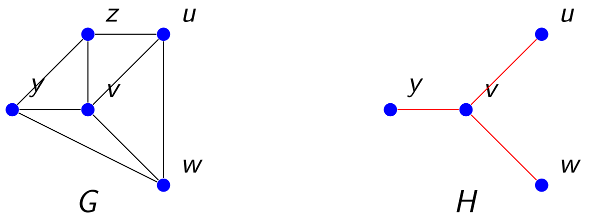
December 18, 2017

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 D of each other in H.
- 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

- Solving linear programs is often challenging task in distributed settings.
- Focus has mainly been solving packing and covering linear programs in a distributed manner (e.g. Kuhn, Moscibroda, and Wattenhofer 2006).

We provide efficient distributed algorithms for distance-bounded network design LPs and a class of convex problem that generalize these problems.

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 $\mathcal{LOCAL}$ model, and local computation is polynomial time.

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 $\mathcal{LOCAL}$ model, and local computation is polynomial time.

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 $\mathcal{LOCAL}$ model, and local computation is polynomial time.

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 $\mathcal{LOCAL}$ model, and local computation is polynomial time.

Communication model

- $\mathcal{LOCAL}$ model:
  - Nodes take step in synchronous rounds and in each round every node can send an arbitrary message of unbounded size to each of its neighbors in the underlying graph $G = (V, E)$.
- Communication is bidirectional, but the input graph may be directed.

Related Work

- Kuhn, Moscibroda, and Wattenhofer (2006) use a Linial-Saks decomposition to solve a packing or covering LP.
- Dinitz and Krauthgamer (2011) showed how to solve the Basic 2-Spanner LP in $O(\log^2 n)$ rounds.
    - We use similar techniques based on padded decompositions.
- Barenboim, Elkin, and Gavoille (2016) showed for any integer parameters k, α, gives an $O(n^{1/\alpha})$-approximation for Directed k-Spanner in $exp(O(\alpha)) + O(k)$ time.
    - They require heavy (exponential time) local computation.

Related Work

- Kuhn, Moscibroda, and Wattenhofer (2006) use a Linial-Saks decomposition to solve a packing or covering LP.
- Dinitz and Krauthgamer (2011) showed how to solve the Basic 2-Spanner LP in $O(\log^2 n)$ rounds.
  - We use similar techniques based on padded decompositions.
- Barenboim, Elkin, and Gavoille (2016) showed for any integer parameters k, α, gives an $O(n^{1/\alpha})$-approximation for Directed k-Spanner in $exp(O(\alpha)) + O(k)$ time.
  - They require heavy (exponential time) local computation.

Related Work

- Kuhn, Moscibroda, and Wattenhofer (2006) use a Linial-Saks decomposition to solve a packing or covering LP.
- Dinitz and Krauthgamer (2011) showed how to solve the Basic 2-Spanner LP in $O(\log^2 n)$ rounds.
  - We use similar techniques based on padded decompositions.
- Barenboim, Elkin, and Gavoille (2016) showed for any integer parameters k, α, gives an $O(n^{1/\alpha})$-approximation for Directed k-Spanner in $exp(O(\alpha)) + O(k)$ time.
  - They require heavy (exponential time) local computation.

Approximation with Linear Programming

- The discrete problem is modelled via an integer program of the following form:

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x = \{0,1\}^d \end{aligned}$$

- The problem is then relaxed into a linear program by changing each integer constraint $x_i = \{0,1\}$ to to $0 \leq x_i \leq 1$.
- Fractional solutions will be rounded to integers using an appropriate rounding scheme.

Distance-Bounded Network Design Convex Program

- We consider convex programs of the following form:

$$\begin{aligned}
\min \quad & g(x) & \\
\text{s.t.} \quad & \sum_{P \in \mathcal{P}_{u,v} : e \in P} f_P \le x_e & \forall (u,v) \in \mathcal{S}, \forall e \in E \\
& \sum_{P \in \mathcal{P}_{u,v}} f_P \ge 1 & \forall (u,v) \in \mathcal{S} \\
& x_e \ge 0 & \forall e \in E \\
& f_P \ge 0 & \forall (u,v) \in \mathcal{S}, \forall P \in \mathcal{P}_{u,v}
\end{aligned}$$

- $g(x)$ is typically a linear function, but more generally a convex function that has a certain partitionability property.
- $\mathcal{P}_{u,v}$ is a set of allowed paths, we assume that length of these paths are bounded by D.

Solving the Distance-Bounded Network Design Convex Program

- Our main result is the following:

Theorem

For any constant $\epsilon > 0$, any distance-bounded network design convex program can be solved up to a $(1+\epsilon)$-approximation in $O(D \log n)$ rounds in the $\mathcal{LOCAL}$ model, where $D = \max_{(u,v)\in\mathcal{S}} \max_{P \in P_{u,v}} \ell(P)$.

- If the convex program can be solved in polynomial time in the centralized sequential setting, then the distributed algorithm uses only polynomial-time computations at every node.

Padded Decomposition

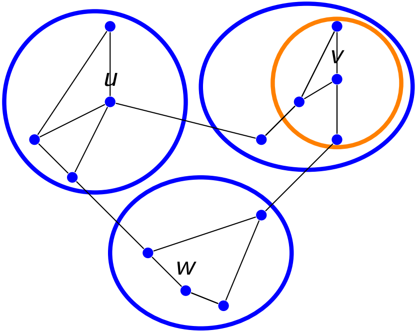
Definition

Given an *undirected* graph G, a (k, ϵ)-padded decomposition, where $0 < \epsilon \leq 1$, is a probability measure μ over the set of graph partitions (clusterings) that has the following properties:

1) For every $P \in \text{supp}(\mu)$, and every cluster $C \in P$, we have: $diam(C) \leq O((k/\epsilon) \log n)$.
2) For every $u \in V$, the probability that all nodes in $B(u, k)$ are in the same cluster is at least $1 - \epsilon$.

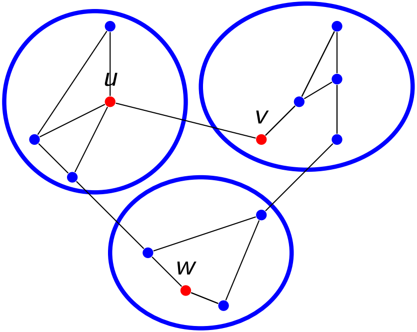
Padded Decomposition

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



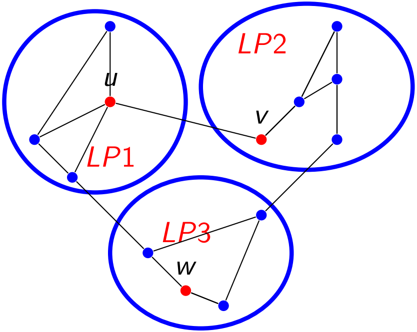
High Level Idea: Partition

- 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.

Applications

- Distance-bounded network design problems that have a local rounding can be solved.
- With high probability there is an $O(n^{1/2} \ln n)$-approximation to the Directed-k-spanner problem that runs in $O(k \log n)$ time in the $\mathcal{LOCAL}$ model.
- There is a distributed algorithm that w.h.p. computes an $\tilde{O}(\Delta^{(1-1/k)^2})$-approximation to the Lowest-k-Degree Spanner problem, taking $O(k \log n)$ rounds of the $\mathcal{LOCAL}$.

- Thanks!
- Questions?