SODA 24

A brief introduction to some interesting papers

Zhiwei Ying

2024-08-07

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目录

1 **Edge-disjoint paths in expanders: online with removals**

2 Fully Dynamic Consistent k-Center Clustering

Preliminary

Given a d-regular expander graph G with n vertices ((n,d)-graph), several pairs of vertices are in this graph. The goal is to find edge-disjoint paths between pairs of vertices.

Preliminary

SHANGHAI JIAO TONG UNIVERSITY

Given a d-regular expander graph G with n vertices ((n, d)-graph), several pairs of vertices are in this graph. The goal is to find edge-disjoint paths between pairs of vertices.

A property on d-regular graph

Given a d-regular graph G and its adjacency matrix A_G, d is the maximal eigenvalue of A_G.

Preliminary

Given a d-regular expander graph G with n vertices ((n, d)-graph), several pairs of vertices are in this graph. The goal is to find edge-disjoint paths between pairs of vertices.

A property on d-regular graph

Given a d-regular graph G and its adjacency matrix A_G, d is the maximal eigenvalue of A_G.

An (n, d, λ)-graph is a d-regular graph with n vertices and the second largest absolute eigenvalue of its adjacency matrix is at most λ.

Preliminary

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

For a given $S \subseteq V(G)$, denote with $e_G(S)$ the number of edges with both endpoints in S. A (n, d)-graph is a (β, γ)-expander if $\forall S \subseteq V(G), |S| \leq \frac{n}{2}$,

$$e_G(S) \leq \begin{cases} \gamma d|S|, & |S| \leq \beta n \\ \frac{d|S|}{3}, & \beta n \leq |S| \leq \frac{n}{2} \end{cases} . \tag{1}$$

Settings

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

Given a (n, d)-graph G and $r \in \mathbb{N}$, there are 2 players: Router & Adversary.
Router maintains a family of pairwise edge-disjoint paths $\mathcal{P}$.
In each turn, there are 2 choices for the adversary. Request 1 is to find a path from a to b that is edge-disjoint from all the other paths in $\mathcal{P}$, and a, b appears less than $\frac{d}{200}$ paths in $\mathcal{P}$. If the path does not exist, the router loses the game. Request 2 is to remove a path P from $\mathcal{P}$.

Observation

Let G be an (n, d, λ)-graph, the diameter of G is at most $\lceil \log(n-1)/\log(d/\lambda) \rceil$.
So if we can nicely distribute the paths, since there are $\frac{nd}{2}$ edges, $r = O(nd\frac{\log(d/\lambda)}{\log n})$.

Previous Results

SHANGHAI JIAO TONG UNIVERSITY

1. Improve r to $\Theta(n/\log^C n)$, for some $C \geq 7$.
2. Propose a randomized polynomial time algorithm that connects any pair of $\Theta(nd/\log n)$ pairs of vertices, as long as each vertex appears as an endpoint at most εd times, for some constant $\varepsilon > 0$.
3. Design a deterministic polynomial-time algorithm that could handle $\Theta(nd/\log n)$ requested pairs and operate in an online manner.

Result

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

1. Let G be an (n, d) (β, γ)-expander graph, for $200 < d < n$, positive constants and $\gamma < 1/1000$, and n large enough. Then Router has a strategy to win the r-Routing game for $r = \varepsilon nd/\log n$, for a constant $\varepsilon = \varepsilon(\beta, \gamma) > 0$. Moreover, the Router can respond to each request in $O(n^3 d^3)$ time and each path in $\mathcal{P}$ is of length $O(\log n)$.
2. Let G be an (n, d, λ)-graph for $200 < d < n$, where $\lambda < \varepsilon d$ for a small enough constant $\varepsilon > 0$. Then Router has a strategy to win the r-Routing game for $r = \alpha nd \frac{\log(d/\lambda)}{\log n}$, for some absolute constant $\alpha > 0$. Moreover, Router can respond to each request in $O(n^3 d^3)$ time and each path in $\mathcal{P}$ is of length $O(\frac{\log n}{\log(d/\lambda)})$.

Future Work

1. Improve time complexity.
2. Optimize the constants.
3. Handle non-regular graphs. In this paper, the writer claims that their algorithm can be applied to almost regular graphs (each degree is within a small range around d).

Basic Setting: k-center clustering problem

Given a metric space $(\mathcal{X}, d)$ and a set of points $P \subseteq \mathcal{X}$, the k-center problem is to output a set $C \subseteq \mathcal{X}$ of at most k centers such that the maximum distance of any point $p \in P$ to the nearest center $c \in C$ is minimized; in other words, the goal is to minimize $\max_{p \in P} \min_{c \in C} d(p, c)$.

Setting: consistent k-center clustering problem

Goal: to maintain a constant factor approximate k-center solution during a sequence of n point insertions and deletions while minimizing the *recourse*, i.e., the number of changes made to the set of centers after each point insertion or deletion.

Previous Result

SHANGHAI JIAO TONG UNIVERSITY

1. In the offline setting, k-center admits several well-known greedy 2-approximation algorithms. It is known to be NP-hard to approximate the objective to within a factor of $(2-\varepsilon)$ for any constant $\varepsilon > 0$.
2. In the incremental setting, where points are inserted but not deleted, one can maintain a $O(1)$-approximate solution for k-center with $O((k \log n)/n)$ amortized recourse.

Result

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

There exists a fully dynamic deterministic algorithm that maintains a constant-approximate solution of the k-center problem and obtains worst-case recourse of at most 1 for an insertion and 2 for a deletion.

谢 谢